

密级: 保密期限:

北京郵電大學

硕士学位论文



题目: 基于 Kinect 的抠像算法研究与应用

学 号: 2013140088

姓 名: 楼珊珊

专 业: 电子与通信工程

导 师: 李学明

学 院: 信息与通信工程学院

2015 年 11 月 25 日

独创性（或创新性）声明

本人声明所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京邮电大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

本人签名：_____ 日期：_____

关于论文使用授权的说明

学位论文作者完全了解北京邮电大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属北京邮电大学。学校有权保留并向国家有关部门或机构递交论文的复印件和磁盘，允许学位论文被查阅和借阅；学校可以公布学位论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存、汇编学位论文。（保密的学位论文在解密后遵守此规定）

保密论文注释：本学位论文属于保密在____年解密后适用本授权书。非保密论文注释：本学位论文不属于保密范围，适用本授权书。

本人签名：_____ 日期：_____

导师签名：_____ 日期：_____

基于 Kinect 的抠像算法研究与应用

摘要

随着多媒体技术的发展和数字传媒业的科技化，数字抠像和合成作为数字图像处理领域炙手可热的研究课题，在电影、游戏和广告制作业，以及医疗卫生、三维重建和空间探测等领域都有广泛的应用。

数字抠像技术是指通过算法在图像和视频中提取带有透明度的兴趣前景。作为一个欠约束问题，传统的自然抠像算法大都需要用户提供划分好的三元图或涂鸦信息作为提示信息，需要大量的时间和人力成本，同时使得抠像算法很难应用于视频抠像。随着深度信息获取技术的发展，一系列结合深度信息的 RGBD 抠像算法应运而出。本文主要研究基于 Kinect 深度信息的自然图像抠像算法，最终实现无需人工输入辅助信息的全自动抠像系统。论文主要完成工作如下：

(1) 研究了体感外设 Kinect 的起源、发展和应用，比较并总结了一代和二代传感器的硬件配置和深度图像获取技术。针对 Kinect 二代获取的深度图像存在的误差，将结合深度信息的 RGB-D 引导滤波引入到深度图像的修复中，并将其进行迭代计算以提高对深度图边缘和内部空洞的修复精度。

(2) 分析及对比了几种传统自然图像抠像技术的原理和应用场景，并深入研究了基于 TOF 的深度抠像算法，将 Kinect 获取的深度信息引入传统的 RGB 抠像算法中。综合考虑抠像的实时性和精度，选择了 Shared Matting 作为本文研究算法。

(3) 实现了三元图的自动生成算法，并将深度信息引入 Shared Matting，提出了基于深度域改进区域扩张方法。最终实现了基于 Kinect 深度信息的全自动抠像算法，经测试该算法在鲁棒性和实时性能得到令人满意的效果。

(4) 设计并搭建了带有可视化界面的 Kinect 实时抠像和合成系统。针对抠像算法中的人物前景与新的虚拟背景之间存在的色调和光照条件的不一致，在系统中引入保持色调一致性的 Erik Reinhard 色调迁移算法。最终实现了 Kinect 彩色视频流的实时人体抠像和合成，并能对前后景色调一致的新的“人景合一”的图像进行存储。

关键词：Kinect 深度信息 Shared Matting 三元图 引导滤波

APPLICATION RESEARCH ON MATTING ALGORITHM BASED ON KINECT

ABSTRACT

With the development of multimedia technology and the technicalization of digital media industry, digital matting and composition, which have a wide range of applications in movies, games and advertising production, as well as medical health care, 3D reconstruction and space detection, become more and more important and are hot topics in research of digital image processing field.

Digital image matting technology is used to extract the interesting foreground with transparency. As an ill-posed problem, most traditional natural image matting algorithm requires the user to provide a trimap or other auxiliary information as a reminder, which costs a lot of time and manpower, making it hard to be applied to the video matting. With the development of the acquisition technology of depth information, a series of RGBD matting algorithms combine the depth information are proposed. In this thesis, we mainly studied the natural image matting algorithm based on depth information captured by Kinect, and realized the automatic Kinect matting system without any manual auxiliary information. The main achievements of this thesis are listed as follows:

(1) This thesis studied the origin, development and application of Kinect sensor, compared and summarized the difference of hardware configuration and depth image acquisition technology between Kinect v1 and Kinect v2 sensor. To deal with the error in the depth image captured by Kinect v2, we combined the depth information with the guided filter as a RGB-D filter. What's more, we obtained the depth information iteratively to improve its precision.

(2) We made the analysis and comparison of several traditional natural image matting technology, especially the matting algorithms based on the

depth information captured by TOF camera. This thesis combined the traditional RGB image matting algorithm with the depth information captured by Kinect sensor, and chose Shared matting as the main algorithm to ensure the real-time and accuracy of this image matting algorithm.

(3)We realized the automatically generation method of trimap, and combined the depth information with Shared matting, proposed a regional expansion method based on depth information. This thesis achieved the automatic image matting algorithm based on depth information captured by Kinect sensor, which can perform in real-time and robustly.

(4)This thesis designed and achieved a visual real-time matting and composition system based on Kinect. To deal with the difference of tonal and light conditions between foreground person and new virtual background, we achieved the color transfer algorithm of Erik Reinhard in the system. This system can accurately extract the person from the real background and compose it with the new virtual background in real time, which has the same tonal with the new background. And we can save the new composed image finally.

KEY WORDS: kinect depth information shared matting trimap
guide filter

目 录

第一章 绪论	1
1.1 研究背景和意义.....	1
1.1.1 图像处理中的数字抠像.....	1
1.1.2 Kinect 与数字抠像.....	2
1.2 国内外研究现状.....	3
1.2.1 数字图像技术的发展.....	3
1.2.2 基于深度图的抠像技术.....	6
1.3 论文的主要内容与章节安排.....	7
1.3.1 本文主要研究内容.....	7
1.3.2 本文的章节安排.....	8
第二章 深度成像技术与抠像算法	9
2.1 深度图像的获取.....	9
2.1.1 双目测距技术.....	9
2.1.2 结构光技术.....	10
2.1.3 飞行时间（TOF）技术	11
2.2 抠像算法基本原理.....	11
2.3 传统经典的抠像算法.....	12
2.3.1 基于颜色采样的抠像算法.....	12
2.3.2 基于传播的抠像方法.....	15
2.3.3 结合颜色采样与传播的抠像方法.....	16
2.4 基于 TOF 相机的深度抠像算法	20
2.4.1 结合贝叶斯抠像和泊松抠像的 TOF 深度抠像算法	20
2.4.2 结合 Robust Matting 的 TOF 深度抠像算法	22
2.5 本章小结.....	24
第三章 基于 Kinect 的抠像算法研究	25
3.1 kinect 工作原理.....	25
3.1.1 Kinect 简介	25
3.1.2 Kinect 深度图	30
3.1.3 深度图与彩色图的坐标映射.....	34
3.2 Kinect 深度图的平滑和滤波	37
3.2.1 引导滤波的基本原理.....	37

3.2.2 基于深度图像的引导滤波.....	39
3.2.3 引导滤波的迭代使用.....	42
3.3 基于 Kinect 深度的三元图自动生成.....	42
3.3.1 数字抠像中的三元图.....	42
3.3.2 三元图的自动生成.....	43
3.4 基于 Shared matting 的抠像算法	44
3.4.1 基于深度改进的区域扩张.....	45
3.4.2 样本采集优化与 Mask 计算.....	46
3.4.3 局部平滑.....	48
3.5 基于 Kinect 的抠像算法测试与分析	50
3.5.1 算法流程.....	50
3.5.2 算法平台与测试结果.....	51
3.6 本章小结.....	58
第四章 基于 Kinect 的抠像系统的设计与实现.....	59
4.1 系统组成与工作流程.....	59
4.1.1 系统介绍.....	59
4.1.2 系统架构.....	59
4.2 系统的开发环境.....	60
4.2.1 Kinect for Windows SDK	60
4.2.2 OpenCV	61
4.3 Kinect 彩色图像与深度图像的获取.....	61
4.4 抠像算法的实现.....	64
4.4.1 引导滤波的 OpenCV 实现	64
4.4.2 三元图的自动生成.....	66
4.4.3 基于深度改进的 Shared Matting.....	67
4.5 色彩匹配算法的实现.....	67
4.6 图像的合成与存储.....	69
4.7 实验结果与分析.....	70
4.8 本章小结.....	73
第五章 总结与展望	75
5.1 本文工作总结.....	75
5.2 不足与展望.....	75
参考文献	77
致 谢	80

第一章 绪论

1.1 研究背景和意义

随着科技和多媒体技术的发展，家用数字产品在人们的生活中日益普及，人们对于多媒体技术和数字获取技术有着越来越深入的研究和掌握。而数字传媒业伴随着影视业和游戏业的科技化，虚拟现实和增强现实技术等图像处理技术得到越来越多科研爱好者和开发者的重视。

作为图像处理领域中的研究热点，数字抠像和合成技术已经越来越多的参与到人们的日常生活中，其在景点景区拍照纪念、影楼、游乐园、科技馆和展览馆等场所都有着重要的应用价值。而本课题所研究的基于 Kinect 的 RGBD 抠像算法，在传统的 RGB 抠像算法的基础上引入了 Kinect 获取的深度信息数据，大大的减少了传统抠像算法所需要的人工交互的工作量并突破了外部条件对图像获取技术的限制和约束。

1.1.1 图像处理中的数字抠像

通过数字抠像技术从图像或视频中将感兴趣的前景从背景中精确地提取出来，是计算机视觉领域的重要问题。其在军用，商用和民用等各个方面都有着重重要的应用价值，不仅是传统的电影和广告制作业，还包括医疗卫生、工农业、环境监测、三维重建、空间探测和电脑游戏制作等等领域。

从早期的“蓝屏抠像”到如今的自然抠像，从数字抠像技术的发展历史来看，电影、游戏和传媒行业的发展以及其在计算机视觉领域的应用成为了推动其发展的重要力量。数字抠像技术可以将现实场景与虚拟艺术创造完美结合，以达到虚拟现实以及增强现实的效果。在电影的后期特效制作中，借助数字抠像技术，科幻电影能够轻易将真实镜头前的演员表演和计算机生成的虚拟场景无缝融合；除此之外，电影特效中的增强现实技术还能将虚拟对象添加到真实环境中，并且所加入的虚拟对象能随着真实环境的角度、透视和光照条件的变化进行调整。《阿凡达》、《哈利波特》、《侏罗纪世界》和《少年派》等科幻电影中惊人震撼的视觉效果都依赖于数字抠像技术和图像合成技术的完美融合。在游戏产业中，越来越多的“浸入式”体验游戏伴随着游戏设备形式的丰富化也逐渐发展起来，借助完美的抠像技术，人们可以轻易地与游戏中的角色进行互动，甚至在同一个场景中“战斗”。在传媒行业中，越来越多美轮美奂的舞台效果借助于实时抠像技术来完成，在 2015 年的春节联欢晚会上，李宇春的歌唱节目《蜀绣》就结合了数字

抠像技术和全息投影等增强现实的技术，实现了舞台中演员真人和背景效果的无缝融合和切换，给全国观众带来了新奇震撼的视觉盛宴。因此，数字抠像技术的商业价值可见一斑。同时，伴随着人们生活水平的提高，数字抠像技术在互联网媒体传播、日常音视频后期处理等方面也得到了越来越广泛的应用。

此外，数字抠像技术在医学、环境学、工农业和教育行业上也有巨大的科研价值和商业价值。在医学领域中一些疾病的诊断需要提取核磁共振图像中的一些特定器官以突出病灶区域；在环境和农业领域中，通过对卫星图像的抠像处理，观察地形、山脉以及农作物从而分析检测地形、海岸线绘制、农作物产量以及生长态势等等；在新型的在教育行业中，将传统的教育方式与沉浸式的体验教育结合，学生可以更加身临其境地体验和学习各种科技知识。因此，数字抠像技术在各传统制作和新兴科研领域中都有着巨大的应用价值。

由于传统的基于图像的抠像算法大多需要人工输入已经划分好的三元图或是涂鸦辅助输入，其对于图片和视频的拍摄环境也有着较高的要求，其效果很容易受到光照、时空等外界因素的影响。因此近年来，随着深度信息获取设备的发展，研究人员开始探索和研究利用深度摄像机获取的深度信息结合图像的彩色信息进行数字抠像，从而减少数字抠像中的人工交互工作量和环境的局限性。因此，基于深度信息的数字抠像算法已经得到了越来越多的重视，有着重大的研究价值和意义。

1.1.2 Kinect 与数字抠像

Kinect 是 Microsoft (微软) 公司在 2010 年 6 月 14 日发布的 XBOX360 体感周边外设^[1]。Kinect 一发布便颠覆了传统游戏的单一操作模式，成为一个革命性的产品，使得“人机交互”的理念彻底展现出来，为很多游戏爱好者带来了非凡的游戏体验。Kinect 作为一种 3D 体感摄影机，结合了即时动态捕捉、图像识别、手势识别、麦克风阵列输入和语音识别等功能，玩家不仅可以体验微软推出的 Kinect 配套游戏《星球大战》以及各种跳舞游戏、运动游戏和冒险游戏等等，也可以轻松地实现网络社交功能，与其他用户进行互动分享。而 2011 年 6 月微软正式推出了 Kinect for windows SDK Beta，并在此后不断更新 SDK 开发包，其可供开发者直接使用 Kinect 进行研究和开发，引起了广大开发爱好者的兴趣。而 2013 年 5 月 28 日，微软又发布了新一代 Kinect 作为 XBOX One 主机的体感外设。研究者可以使用 Kinect V2 获取和识别的语音、手势和人体骨骼信息来进行感知和进一步开发，从而带来更强大的互动式体验。

目前 Kinect 的相关技术已经被广泛的应用于医疗复建、商场购物、教育培训、娱乐游戏以及展会等领域。其独特的 RGB-D 图像获取技术也为数字抠像领

域的发展提供了更强大的研究条件。

第一代 Kinect 采用 Prime Sense 公司开发的光编码 (Light Coding) 技术来获取深度信息，其采用连续照明方式，只需要普通的 CMOS 感光芯片儿而不需要特制的感光芯片，并且其不仅可以以每秒 30 帧的帧率获取彩色信息和深度信息，还能实时捕捉人体的姿态和形体动作，这使得深度信息获取的成本大大降低，在后文中也会提到光编码技术的具体实现原理。而随之而来的二代携带了 TOF (Time of Flight) 技术，相比于之前的结构光技术，TOF 具有更高效的处理速度和更稳定的精度，也为获得更高帧率，更好效果的深度图像提供强大的硬件基础。

传统的数字抠像通常分为蓝屏抠像和自然抠像两大类^[2]。实时抠像系统大多需要借助于“蓝屏”，即其大多应用于虚拟演播室中，采用单色幕布作为人物或是其他对象的背景，通过去除拍摄的实时画面中的幕布颜色来提取完整的前景对象。因此其对于拍摄环境、物体色彩以及人物衣着都有比较严格的要求，并且容易受到光照等外界因素的影响。而在自然抠像中，对于用户所处的环境背景没有严格的要求，然而大部分都需要用户提供提示图像，即将图像分为已知前景、已知背景和未知区域的三元图，而大多三元图需要手工绘制，这不仅增大了人工和耗时，也使得自然抠像算法很难应用于多帧视频图像抠像中。借助于 Kinect 提供的深度信息，我们可以通过深度信息来大致区分人物和背景，因此不会受到光线明暗、人物衣着颜色以及背景颜色的影响，甚至在暗室中仍可以区分前后景，这是传统的机器视觉算法无法比拟的。另外，利用 Kinect 进行人物抠像极其便捷，其不需要任何辅助设备，只要一台 Kinect 设备和一台电脑就可以完成。因此，基于 Kinect 的抠像算法不仅解除了传统图像对于抠像环境的限制，其深度图像获取的简便性和稳定性也为前景图像的提取提供了质量和效率的保证。

1.2 国内外研究现状

1.2.1 数字图像技术的发展

数字抠像 (Foreground Matting) 技术，即图像前景提取技术，起源于图像分割 (Foreground Segmentation) 问题。图像分割作为计算机视觉中图像处理、分析、识别的关键预处理步骤，指的是将图像分成若干个特定的、具有独特性质的区域并将其中感兴趣的目标区域提取出来的技术和过程^[3]。抠像算法与传统图像分割最大的区别在于，其存在着图像背景和前景的重叠区域，即其所得的前景对象中存在着带有透明度的像素，而对于这些透明度像素的处理也是数字抠像技术的主要研究内容和处理难点。图像分割和数字抠像都假定图像 I 包含 N 个像素，即 $I = \{I_1, I_2, I_3, \dots, I_N\}$ ，其中每个像素 I_i 的色值是由对应像素的前景色值 F_i 以及

背景色值 B_i 线性组合而成，即用式（1-1）来表示：

$$C_i = \alpha_i F_i + (1 - \alpha_i) B_i \quad (1-1)$$

其中 $\alpha_i \in [0,1]$ 表示像素 I_i 在所占的前景的不透明度。

在图像分割问题中，每个像素只可能完全属于前景，或完全属于背景，因此对于 α_i 的取值：

$$\alpha_i = \begin{cases} 0 & I_i \in background \\ 1 & I_i \in foreground \end{cases} \quad (1-2)$$

而在实际生活中，绝大多数图像中包含的前景和背景之间会存在半透明的细微物体，比如毛发、塑料以及烟雾等等。这些前景和背景交叠处的半透明像素点的 $\alpha_i \in (0,1)$ ，而传统的图像分割技术无法解决这类细微物体的前景提取问题，因此基于透明度的数字抠像算法应运而生。在上式 1-1 的基础上，对于彩色图像 I ，每个像素的颜色 C_i ，对应的前景值 F_i 以及背景色值 B_i 在 RGB 空间上都为三维向量，因此在 3 个通道中需建立 3 个方程，即：

$$\begin{cases} C_R = \alpha_i F_R + (1 - \alpha_i) B_R \\ C_G = \alpha_i F_G + (1 - \alpha_i) B_G \\ C_B = \alpha_i F_B + (1 - \alpha_i) B_B \end{cases} \quad (1-3)$$

由式 1-3 可知，上述方程组中共有 3 个已知量，7 个未知量，为不定方程，因此图像抠像问题实际上是一个病态（ill-posed）问题^[4]。因此，在解决抠像问题时，通常需要提供额外信息作为补充信息，或是进行先验假设以添加限制从而进行上述方程组的求解。

在早期研究中，为减少上述方程组的未知量，通常会将需抠像的前景物体置于已知颜色的背景图像前进行采集，这种方法即是“蓝屏抠像”。早在 1996 年，Smith 等研究者就提出了一种三角抠像法^[5]，获取多幅不同已知颜色背景下的同一前景对象，已知的背景增加了上述抠像方程组中的已知量，使得方程组有确定的解，从而解决前景抠像问题。由于三角抠像法将抠像问题转化为超定方程的求解，可使用最小平方框架来计算并能达到较为理想的效果，因此该方法目前常被用于生成标准的抠像结果（Ground truth）^[6]，即作为算法处理结果测试和评价的依据。而由于蓝屏抠像需要特定的背景环境辅助，因此目前多用于演播室等较为固定的环境中，实际生活中的应用范围不广。

近年来，自然抠像成为一个集中的研究热点，由于数字抠像本身的病态特性，因此需要用户输入辅助信息，在现有的算法中，通常是通过用户人工交互的方式来获取，而主流的交互方式有三分图和涂鸦输入两种。在实际图像处理中，三分图的创建需要繁琐的人工交互，而且对于图像中的复杂形状，很难得到较为有效

的三分图，因此为提高数字抠像的实用性，涂鸦输入的使用更为广泛。除此之外，在 GrabCut 中，用户还可使用矩形框的形式选中前景图像中的信息，而此方法带来的问题在于前景选取的不准确性，因此需要用户添加一些边界标注。为减少抠像技术中的人工交互，Sun 等人于 2006 年提出了闪光抠像（Flash Matting）^[7]，通过两次拍摄同一场景，将开启和关闭闪光灯获取到的两幅图像作为算法输入，通过计算两者的差值，提取出被闪光的前景物体，从而实现自动抠像的第一步。

自然抠像技术主要有基于颜色采样、基于像素传播和采样传播结合三个大类。

基于颜色采样的抠像方法利用了数字图像中邻近像素在统计特性上的相似性，通过对已知像素进行颜色采样来估算未知区域像素的前景值、背景值以及 α 值。Ruzon 等^[8]最先将概率统计引入到数字抠像中，该算法取未知像素附近的已知前景和背景像素作为样本，进行聚类统计，接着用高斯模型描述每个聚类结果，比较未知区域像素的颜色与已知区域各聚类的相似度，从而推到出该像素的 α 值。在此基础上，Chuang^[9]等人提出了贝叶斯抠像，将贝叶斯公式引入抠像算法，将未知像素的估算转化为了最大后验概率问题。Juan 等人对前景和背景分别使用全局的高斯混合模型（Gaussian mixture model, GMM）进行统计建模^[10]，从而减少贝叶斯统计带来的计算量。然而，该方法无法很好的解决前后景颜色相似或色彩模糊的情况。He 等人在 2011 年提出了全局的采样方法^[11]，将三元图中的所有前景和背景像素视作大小为 $N_F \times N_B$ 的矩阵，将样本点对的采集转化为颜色空间距离上的匹配搜索问题。该方法交替执行扩张搜索与随机搜索进行样本的搜索，有效地避免了样本点对的丢失以及搜索过程中巨大的计算量。

基于像素传播的抠像技术充分利用了邻近像素的相关性，假设一定位置空间距离以及颜色空间距离之内的像素具有一定的规律性，从而使用较少的邻近像素以避免基于颜色采样的方法中由于样本误差导致的错误，通过该方法求解出的前景不透明度在像素间能够平滑过渡，以达到较理想的抠像视觉效果。最典型的泊松抠像^[12]由 Sun 等人于 2004 年在国际图形学会议(SIGGRAPH)中提出，并在基于全局抠像的基础上提出了结合用户交互的局部泊松抠像。Du 等人提取图像的滤波特性作为算法输入，同时用基于该特征的方程替代原有的基于散度的泊松方程^[13]，该方法需要用户提供三元图作为输入，并能得到具有更多图像细节的抠像结果。而随着测地线距离在提取图像中物体形状中的应用，Bai 等提出了基于测地线的抠像框架^[14]。2008 年，Rhemann 等提出了将点扩散函数（PSF）应用于数字抠像^[15]，并在 2010 年的国际计算机视觉与模式识别会议论文中对该方法进行了优化。

结合颜色采样和传播的方法综合了上述两类抠像方法的优点。Wang 等人在 2007 年提出了 Robust Matting 算法^[16]，首次提出了“信任系数”这一概念，通过

信任系数来选择未知像素的前景/背景样本点对，该方法能获取距离未知样本点更远且更多种类的样本，且样本对的颜色与未知像素更为接近，但是其在未知区域过宽且边缘处的样本点通常不具有代表性。Rhemann 等提出使用测地线距离来辅助反映图像的形状信息，从而得到样本点对集^[17]。但是由于其需要遍历更大的空间以获取位置更远的像素点信息，因此其空间复杂度较大。2010 年，Gastal 等提出了 Shared Matting 算法^[18]，通过相邻像素共享样本点对来减少计算量，这也是第一个可以利用 GPU 达到实时抠像的算法。

而自然抠像同样延伸出了数字抠像领域的环境抠像、阴影抠像以及视频抠像。环境抠像^[19]是由 Zongker 等人 1999 年提出的，该算法在自然抠像的基础上，保持了原始图像中前景物体对光照的反射和折射信息，通过 $O(\log n)$ 幅分辨率为 $n \times n$ 的图片进行采样，使得在合成图像中仍然具有这些环境特性，该算法的处理速度较快，其在透明物体的抠像^[20]中有着很重要的应用，但无法处理光的色散现象。Chung 等人^[21]在此基础上将采样图像数到至 $O(n)$ ，提出了一种更精确的环境抠像算法，但只能处理无色透明的前景物体，且其时间复杂度和空间复杂度都较高。Matusik 等提出了一种基于可变视点的环境抠像算法^[22]，其采用[21]中的高精度抠像方法获取选定视角下前景的环境掩模，通过这些视点差值得到其他任意视点的环境掩模，该方法还能获取前景物体的 3D 几何信息。阴影抠像针对的是原始图像中的阴影部分，去除图像中的阴影或者将阴影合成到新的背景中，其是 Chuang 等人于 2003 年提出^[23]，该方法通过获取原始图像中前景的阴影密度图，粗略地得到目标场景的几何信息后将阴影效果进行合成。阴影抠像要求目标场景较为平坦，因此如何将其应用在更为复杂的场景中仍是一个研究难点。视频抠像可视为一组图像序列抠像的扩展，将视频流分割为前景流、背景流以及掩膜流。Chuang 等人首先提出了将 Bayes 方法和光流计算方法结合进行视频抠图，但是该方法的主要处理对象为固定场景或是仅有水平或垂直方向运动的视频流。Apostoloff 等^[24]提出了一种基于贝叶斯框架的全自动视频抠像方法，但该方法的效果在前景较为复杂的视频流中有待提高。今后的相关研究中视频抠像技术会更多地结合物体识别跟踪以及运动分割的技术以达到更精确的抠像效果与更高效的运算效率。

1.2.2 基于深度图的抠像技术

由于深度图像对于光照变化等外界条件不敏感，因此越来越多的抠像算法引入了深度图像作为辅助输入，深度图可用于自动生成三元图，同时可避免基于彩色图像的抠像算法中前后景相邻像素值相近引起的误差。Wang^[25]等利用采集得到的低分辨率深度图进行差值处理及形态学处理自动得到三元图，并用改进后的

Bayesian 抠像算法进行 α 值的计算, 由于该方法没有考虑到相邻像素之间的关联性, 因此无法得到平滑的抠像效果。Zhu 等^[26]在三元图的基础上, 将深度信息作为像素点 RGB 之外的第四个通道来进行 Closed Matting 的改进, 但由于 Closed Matting 本身存在 α 的错误传播问题, 因此同样无法获取精确的抠图效果。Cho J 等^[27]通过二值化结合形态学操作得到三元图, 并在此基础上改进了 Robust Matting, 在前后景相近的区域采用深度图的二值化结果作为相应的 α 值, 该算法能得到较为精确的抠像结果, 但由于其全局样本点对的选取和优化, 因此其计算复杂度较高。为解决抠像算法计算复杂度高、交互繁琐的问题, 何贝等^[28]提出了一种结合 Kinect 深度图的视频抠像算法, 该算法首先对彩色信息进行改进的区域生长算法获取三元图, 并在此基础上进行前后景样本对的二次筛选, 以减少计算复杂度, 最后结合彩色信息、深度信息和鲁棒算法中的置信度进行掩模图像的加权滤波, 最终得到较为平滑、精确的抠像结果。夏倩等^[29]于 2015 年利用改进的三帧间差分法检测视频中的感兴趣区域, 并引入深度信息对区域生长法提出改进, 最终结合 Shared Matting 来获取精确的视频抠像。因此, 随着深度信息获取的便捷, 越来越多的研究者利用类似 Kinect 的深度相机来辅助进行视频图像的抠像操作, 不仅能够避免繁琐的人工交互, 同时能够提高处理时间并得到高精度的抠像结果。

1.3 论文的主要内容与章节安排

1.3.1 本文主要研究内容

本文的研究课题为基于 Kinect 的抠像算法的研究和应用, 通过对 Kinect 工作原理以及基于深度图的数字抠像算法的学习和研究, 主要围绕以下几点展开:

① 介绍了体感外设 Kinect 的起源、发展和应用, 通过对一代和二代 Kinect 传感器硬件配置和深度图像获取技术的比较和总结确立了本文的研究辅助设备——Kinect 二代。通过 Kinect 的硬件组成和数据流的介绍研究了 Kinect 的深度测量原理以及测量误差的主要来源, 并研究 Kinect 在彩色图和深度图在坐标映射上的不足以及解决方案。

② 对比了几种传统经典抠像算法的原理以及抠像结果, 针对本文研究的基于深度图的抠像算法, 最终根据本文研究的抠像实时性和精度综合考虑选择了 Shared Matting 作为本文研究算法。

③ 研究了深度图像的平滑和滤波算法, 将结合深度图像信息的引导滤波引入到深度图像的处理中, 并将其进行迭代使用以提高边缘和内部空洞处理的精度。

④ 实现了深度信息三元图的自动生成算法, 作为本文算法的输入图像之一,

结合深度域改进的 Shared Matting 算法实现了基于 Kinect 的实时抠像。

⑤ 设计并实现了带有可视化界面的 Kinect 实时抠像和合成系统，针对人物前景与虚拟背景色调不一致的问题，引入 Erik Reinhard 提出的经典的色彩迁移算法进行处理，最终能够得到前后景色调一致的新的“人景合一”的图像并进行保存。

1.3.2 本文的章节安排

本文包含五个章节，每章节的具体内容安排如下：

第一章 绪论。综合论述了本文课题的研究背景和意义，包括数字图像处理中的图像抠像算法在不同的生活和科技领域内的应用，普通彩色图像抠像算法和基于深度图的抠像算法目前的研究现状。并结合 Kinect 在深度图像获取上的优势，阐述了本文基于 Kinect 的抠像算法的应用场景与相关技术。最后介绍了本文的内容和章节安排。

第二章 深度成像技术与抠像算法。本章首先介绍了目前的几种主流的深度图像获取技术，包括双目测距技术、结构光技术以及 TOF 技术。根据抠像算法的分类，阐述了三种传统经典抠像算法的基本原理：基于颜色采样、基于像素传播以及结合两者优势的抠像算法。最后结合本文的研究内容重点阐述了基于 TOF 相机的深度抠像算法。

第三章 基于 Kinect 的抠像算法研究。首先介绍 Kinect 的工作原理，就一代 Kinect 与二代 Kinect 传感器硬件和各数据流的特性进行了比较和总结，同时介绍了 Kinect 获取深度图的测量原理、主要误差来源以及深度图与彩色图之间的坐标映射。第二小节结合改进的基于深度图像的引导滤波以及滤波的迭代使用对 Kinect 深度图的平滑与滤波进行了算法研究以及实验结果展示。第三小节介绍了本文提出的基于 Kinect 深度信息的三元图自动生成算法，为本文的实时抠像系统打好基础。第四小节详细阐述了基于深度改进的 Shared Matting 的抠像算法流程，最终对本文基于 Kinect 的抠像算法进行了测试和结果分析。

第四章 基于 Kinect 的抠像系统的设计与实现。本章从系统架构和工作流程展开，结合本文的系统开发环境介绍了基于 Kinect 的抠像系统的设计与实现，对于每个功能模块进行了算法和实现架构上的介绍，最终展示了实现结果并进行分析，论述本文基于 Kinect 的抠像算法在实时抠像系统中的实验结果。

第五章 总结与展望。总结了本文目前的工作成果，并对今后在抠像算法实时性的提高以及边缘平滑度的精度上提出了下一步的研究方向和展望。

第二章 深度成像技术与抠像算法

本章主要介绍了几种深度成像技术，包括双目测距技术，结构光技术和 TOF 技术；同时阐述了几类抠像算法的理论知识，最后介绍了几种基于 TOF 相机的深度抠像算法，并结合本文中的 Kinect 传感器进行说明。

2.1 深度图像的获取

传统的深度图像获取技术主要分为被动测距技术和主动测距技术^[30]。被动测距技术是指通过分析传感器接收到的场景中物体发射或反射的波长信号生成的图像数据结合特定的数学模型来获取该场景的深度信息。主动测距技术则是由系统本身发射的电磁波或声波等来“扫描”场景物体，通过分析计算场景中物体发回的反射光以得到场景的深度图像。常见的被动测距技术有双目测距技术，而常见的主动测距技术主要包括结构光测距以及飞行时间技术（Time of Flight，即 TOF）等，因此本小节将会着重对这三种技术进行阐述。

2.1.1 双目测距技术

双目测距技术^[31]基于人眼利用双眼接收外界光线在视网膜成像从而感知外界事物远近的原理。典型的双目视觉测距系统利用两个普通相机（彩色或是黑白相机）拍摄同一场景，通过两幅图像的像素对应来计算场景的深度信息。其原理如图 2.1。

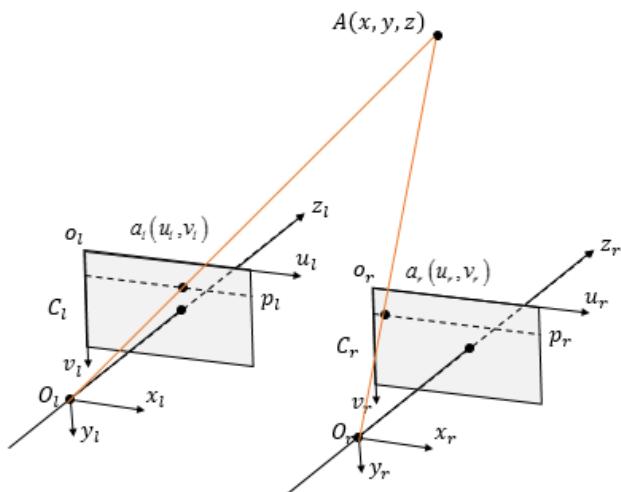


图 2-1 双目视觉测距技术原理

图中 O_l 和 O_r 为两个具有相同焦距 f ，且平行放置的相机， C_l 和 C_r 分别为两

者的成像平面， b 为基线距离，以 O_l 和 O_r 为中心分别建立两个互相平行的空间坐标系，且以 O_l 为主坐标系。场景中有一点 $A(x, y, z)$ ，其在两个平面上的投影分别为 $a_l(u_l, v_l)$ ， $a_r(u_r, v_r)$ ，由于 $v_l = v_r$ ，定义视差 $d = u_l - u_r$ ，由三角形相似得：

$$\frac{b-d}{Z-f} = \frac{b}{Z} \quad (2-1)$$

则：

$$Z = \frac{bf}{d} \quad (2-2)$$

即可得到深度 Z 与视差 d 的关系。而双目测距系统中，已知上式中的基线距离 b ，焦距 f ，只需测量视差 d 即可得到场景中某点的深度值。由上式 (2-2) 可以看出，深度信息与视差 d 成反比，因此当视差较小时，视差的微小变化会导致巨大的深度变化，视差较大时，其大小变化可能会被忽略，导致深度测量值不产生变化。因此，双目测距系统对于场景和相机之间的距离要求不能过远，否则会影响测量精度。除此之外，该系统中视差 d 的计算依赖于场景中的点 A 在两个相机获取的不同视角的图像中的对应的投影，因此，两幅图像像素之间的对应关系成为双目测距系统中的难点。传统方法中，通常对两幅图像进行特征提取和匹配，其前提是该场景图像具有丰富的纹理特征，这在很大程度上限制了双目测距技术在现实场景中的应用。

2.1.2 结构光技术

结构光技术^[32]与双目测距技术的原理相似，都基于三角测量计算。两者不同的是，结构光采用的是主动光源，主动将编码图案投射到场景，通过分析场景反射回来的结构光图案得到与原始图案之间的变换关系，利用三角测量法获取场景中每个像素的深度。虽然原理相似，结构光技术有着较大的优势。第一，结构光技术的光源是主动红外式光源，因此其不易受到场景中光照环境的影响；第二，结构光中的原始编码是开发者设定的，将结构光图案投射到场景中即是对整个场景进行了编码，因此其对于场景中的纹理特征等没有过高的要求，可应用范围较广；第三，在结构光技术中，原始编码模式已知，场景返回的编码图案也很容易得到，因此两者之间的比较匹配难度大大降低，也因此省去了大量的分析和计算步骤。

同样的，结构光技术也存在着不足。首先是当场景中含有与系统的主动光源同频率的光时，该环境光会与光源进行混合，影响最终的测量精确度。太阳光中含有各种频率的光线，因此结构光技术在户外场景中的测量精度较低，应用价值不大。另外，当同一个场景中有多个相同的结构光测距系统进行测量时，各个系统之间的光源会产生干扰，无法得到准确的测量结果。

在 Kinect V1 中，采用的深度测量技术就是结构光技术，而与传统的结构光技术不同的是，Kinect 传感器采用高度随机的三维图像编码代替了原有的周期性变化的二维图像编码。通过 Kinect 发射器的光栅对发射出的红外线形成激光散斑（laser speckle）。测量前，在场景中设固定间隔的多个参考平面记录随机变化的散斑图像作为光源标定，测量过程中通过对红外摄像头获取的测试场景中的激光散斑的变形图像进行视差的测量和计算。得到场景中每个像素点的视差组成的视差图像后，结合 Kinect 数学模型来计算得到场景中每一像素的深度信息。

2.1.3 飞行时间（TOF）技术

飞行时间技术^[33]（Time of Flight, TOF），是指连续向场景发射光脉冲后，测量波源发射出的声波或电磁波，从离开发射器到被场景中的物体反射，最后被接收器接收所花费的时间，即“飞行时间”。由于光波和声波在空气中的传播速度是定值，获取“飞行时间”，即可获取场景到波源之间的距离。

其原理表示如下式：

$$d = \frac{1}{2} \left(n\lambda + \frac{\varphi}{360} \lambda \right) \quad (2-3)$$

其中， λ 为发射信号的波长， n 为信号在飞行时间内经历的波长个数， φ 为返回信号的相位。

飞行时间技术通常采用激光作为发射信号，由于其具有相干性好、方向性好和亮度高等优点，用于精确测量有很大的价值。与结构光技术相比，飞行时间技术的原理更为简单，而其技术重点在于获取极其精确的飞行时间，通常情况下，为满足实际应用所需的精度，大多成熟的飞行时间测距技术都采用原子计时器。

在新一代的 Kinect 设备中，采用主动的 TOF 测距技术进行深度图像的获取，其具体工作原理在第三章将会详细阐述。

飞行时间技术相比于其他测距技术有以下优点：首先，TOF 测距系统不需要双目测距系统中复杂的相关性计算过程，可对场景中的深度信息进行实时计算，达到较高的帧率；其次，TOF 测距技术不受场景物体表面的纹理和色彩的影响，可在多种环境下进行较为准确的三维检测；并且其测量精度较为稳定，对于距离远近的变化有较好的适应性，可应用与一些较大范围的运动场合的深度检测中。

2.2 抠像算法基本原理

根据第一章式 1-1 所述的式子，将数字抠像简化为如下式子：

$$C = \alpha F + (1 - \alpha)B \quad (2-4)$$

其中 C 为已知的原始图像, F 为前景区域, B 为背景区域, α 为抠像最终所得到的掩模图, 即由每个像素点的前景透明度所组成的二值图。在数字抠像算法中, 即是通过各种算法统计和计算, 最终得到由每个像素的前景透明度所组成的图像 α 。

由于本文主要的研究对象为自然图像, 因此在下文主要针对自然图像的传统算法分类进行阐述。

数字抠像方法根据原始图像的色彩类型可分为蓝屏抠像和自然抠像, 而基于自然图像的抠图技术可分为基于颜色采样、基于像素传播和两者结合的方法三大类。自然图像抠像算法的研究进程中, 产生了很多优秀的算法, 2.3 中将从三类方法的原理入手, 详细介绍其实现原理, 并介绍其中的经典算法。

2.3 传统经典的抠像算法

本小节主要从基于颜色采样、基于像素传播和两者结合的三类抠像方法展开经典抠像算法的原理和实现步骤阐述:

2.3.1 基于颜色采样的抠像算法

这类算法基于自然图像在邻近像素上的统计相关性, 具有相似颜色的邻近像素其 α 值也是相似的。根据局部平滑假设, 对每个未知像素附近的前景和背景样本点对进行采样, 估算未知区域像素的前景和背景, 进而得出最优抠像参数 (F, B, α) 。基于颜色采样的抠像算法包括用于 Photoshop 抠像插件的 Knockout 算法, 基于概率求解抠像方程的 Ruzon Tomasi 方法等。而最典型的基于颜色采样的抠像算法是贝叶斯抠像 (Bayes Matting), 该方法基于贝叶斯框架, 其基本原理如图 2-2 所示:

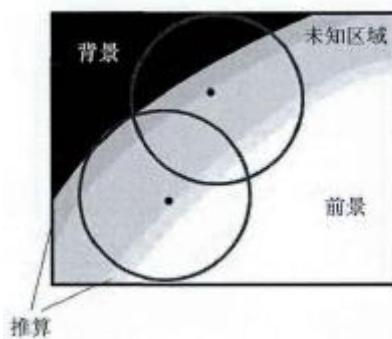


图 2-2 贝叶斯算法 采样示意图

贝叶斯算法假设用户输入的三元图中未知像素的前景样本点和背景样本点

都服从正态分布。如图 2-3 中 A, B 两点, 圆形采样窗口从未知区域的边缘开始向中心采样, 针对不同的未知像素点, 其采样窗口 W 的半径 r 大小可进行调整, 采样过程中, 首先选取较小的 r 作为开始, 然后逐渐扩大, 直到得到足够多的前后景样本对。其次将采样得到的样本点分簇, 并建立各簇的颜色分布, 从而对未知区域的像素点进行颜色估计。通过上述两个步骤, 即将抠像问题转化成了求解最大后验估计的贝叶斯框架问题, 如下图所示:

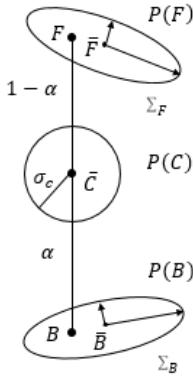


图 2-3 贝叶斯算法 颜色估计示意图

贝叶斯算法的具体描述如下:

对于输入图像, 在确定前景和背景颜色值的前提下, 根据贝叶斯定理和最大后验概率估计未知区域像素的颜色分布:

$$\arg \max_{F,B,\alpha} P(F, B, \alpha | C) = \arg \max_{F,B,\alpha} P(C | F, B, \alpha) P(F) P(B) P(\alpha) / P(C) \quad (2-5)$$

设 $L(*) = \log P(*)$, 则:

$$\text{上式} = \arg \max_{F,B,\alpha} L(C | F, B, \alpha) + L(F) + L(B) + L(\alpha) \quad (2-6)$$

由于 $P(C)$ 为常数, 在求最大值时可直接忽略。

因此贝叶斯求解转化为求解 $L(C | F, B, \alpha)$, $L(F)$, $L(B)$ 以及 $L(\alpha)$

图像未知区域的像素值符合高斯正态分布, 其均值为 $\bar{C} = \alpha F + (1 - \alpha)B$, 标准差为

σ_c , 则:

$$L(C | F, B, \alpha) = -\|C - \alpha F - (1 - \alpha)B\|^2 / \sigma_c^2 \quad (2-7)$$

上式中, 根据每个像素 $N \times N$ 邻域上的已知像素和先前估计得到的像素值来建立颜色概率分布。接着利用图像的空间连续性估计其前景 $L(F)$, 为更好地表示前景的颜色分布, 该算法又为邻域上的每个像素 i 添加了权重 w_i , 有

$$w_i = \alpha_i^2 g_i \quad (2-8)$$

α_i^2 为像素的不透明度, g_i 为标准差 $\sigma = 8$ 时的空间高斯衰减值。

给定图像的前景色和权重后, 使用 Orchard 和 Bouman 方法对颜色进行分簇。对于每个簇, 计算其加权平均颜色 \bar{F} 和加权协方差矩阵 Σ_F :

$$\bar{F} = \frac{1}{W} \sum_{i \in N} w_i F_i \quad (2-9)$$

$$\Sigma_F = \frac{1}{W} \sum_{i \in N} w_i (F_i - \bar{F})(F_i - \bar{F})^T \quad (2-10)$$

其中, $W = \sum_{i \in N} w_i$ 。 $L(F)$ 可建模为椭圆的高斯分布, 则加权协方差矩阵可表示为:

$$L(F) = -(F - \bar{F}) \Sigma_F^{-1} (F - \bar{F}) / 2 \quad (2-11)$$

在自然图像的抠像中, 将 $L(F)$ 中的 F 用 B 代替, 并设置权重 w_i :

$$w_i = (1 - \alpha_i^2) g_i \quad \text{式(2-12)}$$

由于 $L(C | F, B, \alpha)$ 中需求解 F, B, α 三者的成绩, 因此式 2-3 中的最大后验概率求解不是二次方程, 为有效地求解该方程, 将该问题分解为两个二次子方程。对于第一个子方程, 假设 α 为常量, 并将上式中对 F 和 B 分别求偏导, 并求其极大值:

$$\begin{aligned} & \left[\begin{array}{cc} \sum_F^{-1} + I\alpha^2 / \sigma_c^2 & I\alpha(1-\alpha) / \sigma_c^2 \\ I\alpha(1-\alpha) / \sigma_c^2 & \sum_B^{-1} + I(1-\alpha)^2 / \sigma_c^2 \end{array} \right] \begin{bmatrix} F \\ B \end{bmatrix} \\ &= \begin{bmatrix} \sum_F^{-1} \bar{F} + C\alpha / \sigma_c^2 \\ \sum_B^{-1} \bar{B} + C(1-\alpha) / \sigma_c^2 \end{bmatrix} \end{aligned} \quad (2-13)$$

其中, I 表示 3×3 的单位矩阵。因此对于常量 α , 求解上述 6×6 的线性方程即可得到最优参数 F 和 B 。

在第二个子方程中, 假设 F 和 B 为常量, 建立一个 α 的二次方程, 此时借助像素的实际观测颜色 C 来求解颜色空间中的线段 FB :

$$\alpha = \frac{(C - B) \cdot (F - B)}{\|F - B\|^2} \quad (2-14)$$

根据上式, 首先利用常量 α 求解前景颜色 F 和背景颜色 B , 再利用邻域像素 α 的均值初始化本像素的 α 取值, 从而求解式 (2-14)。

当已知像素被分为多于 1 组前后景簇对时, 需对每个前后景簇对进行上述最优化过程, 并求得式 (2-6) 的最大解。

基于像素采样的技术在不同自然图像中的抠像结果很大程度上依赖于辅助

输入的三元图的划分效果以及前景和背景的对比差异。在纹理复杂及前后景色彩过渡不明显的自然图像中，样本的采集结果的误差会造成很大的抠像精度和效果上的影响。

2.3.2 基于传播的抠像方法

相比于基于采样的抠像方法，基于传播的方法不需要对前景和背景的颜色值进行采样估计，其假设图像的前后景满足局部光滑性约束，且在一定位置空间距离和颜色空间距离下，未知像素的邻域有恒定或相似的属性，比如符合线性变化，具有相同的模糊连接度等等，从而在混合参数求解过程中可以避免由于前后景样本点对采集导致的误差，并通过消除前景和背景来获得 α 的闭解。基于传播的方法充分利用了邻域像素的相关性，因此准确获取邻近像素之间的相关性特征，并提高像素之间信息传播的可靠性，是影响最终抠像结果的关键。这类方法中，主要有基于线条交互的闭合解抠像（Closed-form Matting），基于光谱图像分割的全自动抠像 Spectral Matting，泊松抠像（Poisson Matting）等等，除泊松抠像需用户提供三元图外，其他方法都可通过稀疏的涂鸦指示前后景，大大地降低了人机交互的要求，同时也能得到视觉效果较为理想的抠像结果。

基于传播的抠像技术中，最经典的泊松抠像（Poisson Matting）是 Sun 等人于 2004 年提出的。泊松抠像将图像的透明度看作图像本身的一种特征——场，从而将抠像问题转化为求解 α 梯度的泊松方程。在泊松抠像中，对于抠像方程 $C = \alpha F + (1 - \alpha)B$ 进行两边求导，得到如下方程：

$$\nabla C = (F - B) \nabla \alpha + \alpha \nabla F + (1 - \alpha) \nabla B \quad (2-15)$$

其中：

$$\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$$

后续的 α 求解过程分为两步：全局求解和局部优化。

① 全局求解

根据用户输入的辅助提示信息，将需要抠像的自然图像分为已知的前景、背景区域，以及待计算的未知区域。基于邻域像素的相似性，对某一未知像素，分别将前景区域和背景区域中最接近该点的像素点的颜色值作为初始的前景颜色和背景颜色。

由于该算法假设图像中的前景和背景颜色平滑过渡，因此 ∇F 和 ∇B 都可以视为 0，即可忽略式 (2-11) 中的 $\alpha \nabla F + (1 - \alpha) \nabla B$ 。并可得到如下结果：

$$\Delta\alpha = \operatorname{div}\left(\frac{\nabla C}{F - B}\right) \quad (2-16)$$

其中， $\Delta = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)$ 表示拉普拉斯算子， div 为散度。在此基础上，利用 Guass-Sideline 迭代或超松弛迭代，进行式 (3-16) 的求解，并根据 α 的取值将其归于前景 ($\alpha > 0.95$) 或是背景 ($\alpha < 0.05$)，在 α 趋于平稳或是所有像素都划分完毕后，即可获得自然图像的整个掩膜图。

② 局部优化

由于上述基于全局的迭代求解过程前提是图像邻域像素的平滑过渡，因此在一些复杂的自然图像中，全局计算无法取得很好的效果。在此基础上，针对自然图像中局部前景和背景变化迅速的区域 S ，不可直接忽略 $\alpha\nabla F + (1-\alpha)\nabla B$ ，因此可将式 (3-16) 简化为下式：

$$\nabla\alpha = A(\nabla C - D) \quad (2-17)$$

为求解上述问题，Sun 等在全局抠像的基础上提出了局部泊松抠像，即通过用户手动添加局部分区提示，进行区域 S 中复杂颜色像素的 α 值求解。

通过上述两个步骤，在一些复杂自然图像中，泊松抠像可以得到比贝叶斯抠像更好的抠像结果，但是其相应的需要更多的用户输入作为辅助信息，并且该算法的时间复杂度更高。基于这样的问题，后续的泊松抠像优化方法中，将使 (3-16) 中基于散度的泊松方程求解用图像特征值来代替，采用该种方法可省去上述的局部优化过程，并获得更高的抠像效率。

因此，基于传播的抠像方法在复杂颜色的自然图像中可以充分利用邻域像素的相似性假设获得先验知识，将病态的抠像方程转化为有确定解的数学问题，从而获得较好的抠像结果。然而由于没有前后景样本点对的采集过程，该技术的对于不同类型图像的适应性很大程度上受到邻域之间的相似性假设，因此在邻域像素值变化频繁的图像中，该种方法还需进一步的优化和改进。

2.3.3 结合颜色采样与传播的抠像方法

上文阐述了基于颜色采样和基于传播的方法中的经典算法以及各自的优缺点，在数字抠像领域中，一些优秀的抠像方法结合上述两种技术，实现了各类型图像抠像精度和效率的均衡。在这类算法中常用的有：①优化迭代抠图方法^[34]，假设图像掩膜符合马尔可夫随机场 (MRF)，采用迭代优化的方法，同时结合全局颜色采样来引导迭代过程中 α 的传播。②Easy Matting^[35]，该方法同样基于图像中未知区域的马尔可夫随机场假设，通过迭代传播求解能量方程从而进行连续的 α 值估算。在马尔可夫光滑约束项无法很好地代表前后景颜色分布的情况下，

采用局部采样来获取足够的前景和背景样本点对来做抠像结果的精细优化。③鲁棒抠像算法（Robust Matting），该方法在结合采样和传播的方法中最具代表性。本小节将详细介绍 Robust Matting 的原理和其实现技术。

Robust Matting 算法引入了采样过程中的“信任系数”来进行前景和背景样本点对的判断，首先通过采集得到的样本点对对 α 值进行初步的估算，然后由随机漫步最优化一个图标记出初步计算中需要改进的区域，因此其主要实现步骤包括优化的全局样本点对采样和 α 掩膜优化两个步骤。

① 优化的全局颜色采样，获取初步的 α 估计值

Robust Matting 方法同样需要用户在原始图像的基础上输入提示信息作为辅助输入，其提示信息可以是人工分割的三元图，也可以简单的区域涂鸦提示。对于未知区域中的像素，该方法假设其真实的前景和背景颜色总是与样本点对中的部分像素的前景和背景值相近，因此，在求解未知像素时，可通过大量的前景和背景样本点集合来估计该点的前景和背景颜色。

基于采样的抠像方法最常使用最近邻采样法进行样本点采集，即在未知像素的邻域中选择空间距离最小的前景和背景样本点，因此在具有复杂结构的自然图像中无法真实反映前景和背景的真实颜色分布。在 Robust Matting 中，采用了稀疏采样方法，即沿着已知的前景和背景区域的边界稀疏地采集像素样本，使得样本点集能较好地反映前景和背景的变化。

采集得到样本点对后，为更好进行 α 初值的估算，Robust Matting 提出了一种优化的颜色采样方法，即引入了一个“信任系数”，在大量的样本点对中选择置信度更高前景背景样本点对。在传统的投影法中， α 值的初值估计基于以下模型：

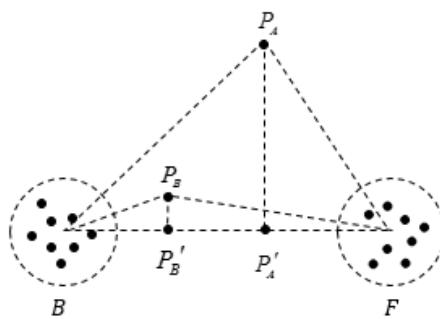


图 2-4 原有抠像算法的线性模型

在上图中， F 和 B 分别代表采集得到的前景和背景样本点集合， P_A 和 P_B 为未知像素点，其在 F 和 B 集合中心点连线上的投影分别为 P'_A 和 P'_B 。在此基础上，根据其投影点和两个样本点集中心之间的关系来建立满足式(2-1)的线性方程，接着进行 α 值的估计。然而投影法中存在如下问题，在上图中 P_A 可很好地用图中的样本点对集合的线性方程表示，但是 P_B 距离两个样本点集的中心点较远，其像

素值由这两个样本点集线性组合而成的可能性很小。为避免上述问题，在 Robust Matting 中，对于某一对前景和背景样本点连线邻近的未知像素，如 P_A ，可直接由这组前景和背景样本点对经过凸组合得到。对于未知像素点 i 和其观测颜色 C ，假设其最优样本点对中前景和背景的样本点颜色分别为 F_i 和 B_i ，则其 α 的估计如下：

$$\hat{\alpha} = \frac{(C - B_i)(F_i - B_i)}{\|F_i - B_i\|} \quad (2-18)$$

定义距离比率 $R_d(F_i, B_i)$ ，其为像素色值 C 与上式估计得到的颜色估计值之间的欧式距离与两个样本点之间欧式距离之比，值越小，说明该样本点对越精确：

$$R_d(F_i, B_i) = \frac{\|C - (\hat{\alpha}F_i + (1-\hat{\alpha})B_i)\|}{\|F_i - B_i\|} \quad (2-19)$$

基于式(2-19)，计算图 2-4 中 P_A 和 P_B 的距离比率， P_B 的计算结果远高于 P_A ，因此 F_i 和 B_i 这组样本点对不适合用来计算 P_B 的 α 值。在上式中，颜色空间中距离较远的样本对，其距离比率越小。而在自然图像中，前景和背景样本点邻域的未知像素原本可能就是一个完全前景像素或背景像素（其 $\alpha=1$ or $\alpha=0$ ），只有很少一部分为半透明像素。如下图所示：

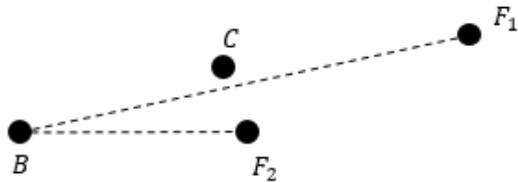


图 2-5 优化颜色采样中距离比率带来的误差

如图 2-5，对于未知像素 C ，若根据距离比率 R_d ，其样本点对应该为 B 和 F_1 。而从颜色空间的欧式距离来说， F_2 与 C 的距离更近，其可能为更好的前景样本点。基于上述情况，只用距离比率 R_d 来判断最优样本点对是不精确的。于是 Robust Matting 引入了每个样本点对的权重值 $w(F_i)$ 和 $w(B_i)$ ：

$$\begin{cases} w(F_i) = \exp\left\{-\|F_i - C\|^2 / D_F^2\right\} \\ w(B_i) = \exp\left\{-\|B_i - C\|^2 / D_B^2\right\} \end{cases} \quad (2-20)$$

其中 D_F 和 D_B 分别表示未知像素 C 与前景和背景样本点之间的最短欧式距离，即 $D_F = \min\{\|F_i - C\|\}$ ， $D_B = \min\{\|B_i - C\|\}$ 。则由式 (2-20) 可知，图像中离未知像素欧式距离越小的样本点，其样本点的权重值越高。

结合距离比率和上述权值，该算法为每个前景和背景样本点对都计算一个信任系数 $f(F_i, B_i)$ ：

$$f(F_i, B_i) = \exp \left\{ -\frac{R_d(F_i, B_i)^2 w(F_i) w(B_i)}{\sigma^2} \right\} \quad (2-21)$$

其中 σ 为误差的标准差，一般设为 0.1。

对于一个未知像素，在采集得到的前景和背景样本点对中，首先根据式（2-18）估算每个样本点对对应的 α 值，同时根据式（2-21）计算该样本点对的信任系数。在此基础上，选择置信度最高的几个样本点对，计算其 α 值和信任系数的均值作为该未知像素的最初 α 值和置信度。这样得到的 α 值和置信度将作为初始值进行后续的 α 掩膜优化。

② α 掩膜优化

Robust Matting 中，通过上述经过优化的采样过程，已经得到了较为精确的 α 值，以及每个像素对应的置信度。在此基础上，加入新的 α 值优先期望进行后续的掩膜优化。首先假设目前得到的 α 掩膜图是局部平滑的，且其完全前景和完全背景的像素数量远大于具有透明度的像素，因此对最终 α 应满足数据约束和邻域约束，即最终求得的 α 值应该与初始的 α 值相关，并且应该是局部平滑且具有图像噪声鲁棒性的。因此将 α 掩膜优化转化为解图标记问题。

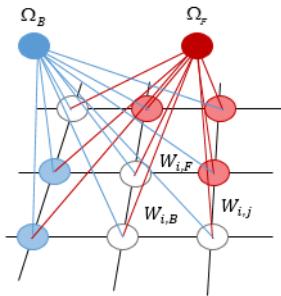


图 2-6 图标记问题优化掩膜

如上图 2-6 所示， Ω_F 和 Ω_B 为两个虚结点，分别表示纯前景和纯背景。白色结点表示图像的未知区域，蓝色结点表示人工标记的背景像素，红色结点表示人工标记的前景像素。为满足数据约束，每个节点与虚节点之间定义了数据权重，其表示该结点属于前景或背景的可能性。对于置信度高的结点，可直接用初始 α 值作为最终的透明度。而对于低置信度的结点，则该像素点可能为完全前景或背景像素。如上图 2-6，对于未知像素 i ，为其赋予与虚结点之间的数据权重 $W(i, F)$ 和 $W(i, B)$ ：

$$\begin{cases} W(i, F) = \gamma \left[\hat{f}_i \hat{\alpha}_i + (1 - \hat{f}_i) \delta(\hat{\alpha}_i > 0.5) \right] \\ W(i, B) = \gamma \left[\hat{f}_i (1 - \hat{\alpha}_i) + (1 - \hat{f}_i) \delta(\hat{\alpha}_i < 0.5) \right] \end{cases} \quad (2-22)$$

其中 $\hat{\alpha}_i$ 和 \hat{f}_i 表示像素 i 的掩膜初始值和置信度。 δ 的取值: $\delta(P) = \begin{cases} 1 & P \text{为真} \\ 0 & P \text{为假} \end{cases}$

γ 为平衡数据权重和边权重的可调参数, 一般设置为 0.1。若其设置过高, 最终的 α 值会被噪声污染, 而其设置过低会导致 α 值过分平滑。

接着定义相邻像素点 i 和 j 的边权重 $W_{i,j}$ 以满足邻域约束:

$$W_{i,j} = \sum_{k|(i,j) \in w_k} \frac{1}{9} \left(1 + (C_i - \mu_k)^T \left(\Sigma_k + \frac{\varepsilon}{9} I \right)^{-1} (C_j - \mu_k) \right) \quad (2-23)$$

上式中, w_k 表示所有包含 i 和 j 的大小为 3×3 的窗口, μ_k 和 Σ_k 则为每个窗口的均值和协方差。 $\varepsilon = 10^{-5}$ 为规范化系数, 可提高计算鲁棒性。

最后利用随机漫步最优化来求解图标记问题, 即可得到最终优化的 α 掩膜图。

首先, 建立如下的拉普拉斯算子矩阵:

$$L_{i,j} = \begin{cases} W_{ii} & i = j \\ -W_{ij} & i, j \text{ 相邻} \\ 0 & otherwise \end{cases} \quad (2-24)$$

其中, L 为稀疏对称的 $N \times N$ 的正定矩阵, N 为图像中结点的总数。接着将分为与已知区域和未知区域对应的 L_k 和 L_u , 即:

$$L = \begin{bmatrix} L_k R \\ R^T L_u \end{bmatrix} \quad (2-25)$$

此时给定一个边界条件向量 m , 即可进行最终 α 掩膜图:

$$L_u = -Rm \quad (2-26)$$

2.4 基于 TOF 相机的深度抠像算法

随着数字抠像技术和深度成像技术的发展, 越来越多的抠像算法引入了对光照变化等外界条件不敏感的深度信息 (Depth) 作为辅助输入信息, 本文将其称为 RGB-D 抠像方法。深度图可用于自动生成三元图, 同时可避免基于彩色图像的抠像算法中前后景相邻像素颜色相近以及复杂的颜色变化引起的误差。本章 2.1 中已经阐述了目前比较经典的深度成像技术, 由于本文研究的基于 Kinect 的抠像算法主要是基于 TOF 成像, 因此本小节主要介绍基于 TOF 的深度抠像算法。

2.4.1 结合贝叶斯抠像和泊松抠像的 TOF 深度抠像算法

Wang 等人最早于[26]中提出了利用 TOF 相机获取的深度图像自动生成三元图的方法, 文中将深度信息作为 RGB 三个色彩通道之外的第四维信息, 对传统

的贝叶斯抠像和泊松抠像算法进行了基于深度图的改进。

在文献中，采用以下三个步骤利用 TOF 获取的深度图像自动生成三元图。第一，上采样。对于同一对象，分别由高清彩色相机和 TOF 相机拍摄获取彩色图像和深度图像，受技术的限制，深度相机获取的深度图像分辨率 64×64 ，远低于彩色图像。而该算法需要输入分辨率一致的彩色图像和深度图像进行处理，因此首先采用上采样^[36]算法提高深度图像的分辨率。第二，二值化。将上采样得到的深度图像进行阈值化操作，从而将其分为前景和背景部分。如图 2-7。第三，形态学腐蚀膨胀。文中通过形态学腐蚀膨胀对于经过二值化的图像边界的未知区域进行处理，从而得到最终包含已知前景、已知背景以及未知区域三种像素的三元图^[37]。其具体操作步骤在后文中会详细阐述。

获得三元图的步骤如下图所示：

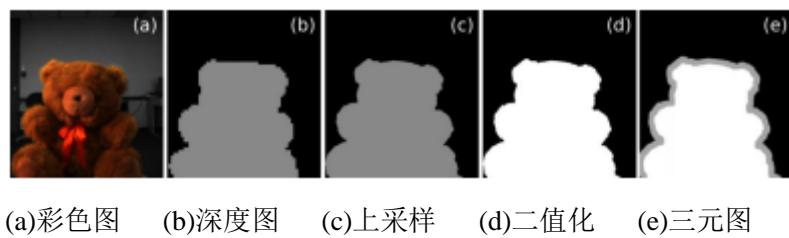


图 2-7 基于 ToF 相机深度图像的三元图

1) 基于 RGB-D 的贝叶斯抠像算法

在贝叶斯抠像中，深度信息可直接对较强边缘的图像区域进行前景或是背景的判断，但对于未知区域，需要借助文中提出的一个逆熵公式：

$$H'(\alpha) = 1 + \alpha \log \alpha + (1 - \alpha) \log(1 - \alpha) \quad (2-27)$$

该函数的值在已知前景和背景区域邻域时，取值较大，因此将其作为一个权值，即可作为深度信息的权值加入到求解贝叶斯框架的抠像过程中。

加入深度信息的 RGB-D 贝叶斯抠像，与传统的基于 RGB 的贝叶斯算法相比，在前景和背景重叠的边缘区域具有更好的抠像结果，如下图所示：

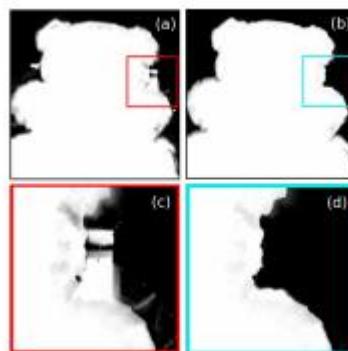


图 2-8 左侧：标准 RGB 贝叶斯抠像 右侧：基于 RGB-D 的贝叶斯算法^[26]

2) 基于 RGB-D 的泊松抠像算法

与贝叶斯抠像相似，在泊松抠像中，将最终的透明度 α 值看做初步估计得到的透明度 $\hat{\alpha}$ （见本文 2.3.2）和二值化后的深度信息 D 的线性组合，并将全局求解之后的置信度 f 作为权值，即对于 f 过低的像素，用其深度信息进行透明度补偿：

$$\alpha = f\hat{\alpha} + (1-f)D \quad (2-28)$$

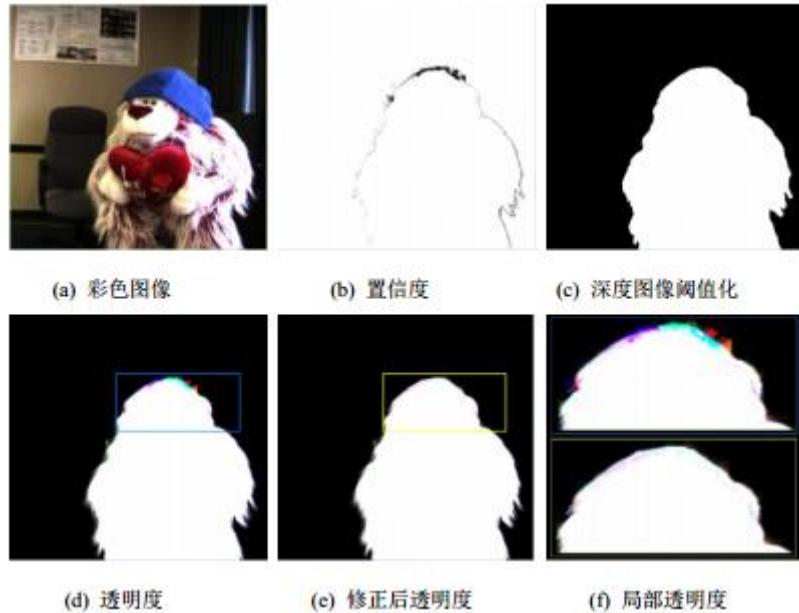


图 2-9 基于 ToF 深度信息的泊松抠像^[26]

如图 2-9 所示，(b) 和 (d) 为泊松方法得到置信度图像和透明度图像，对于置信度图像中的黑色部分，即为置信度低的区域，采用二值化后的深度图像 (c) 进行补偿，最终根据式 (2-28) 对初始的透明度图像进行修正，得到 (e)。

而该方法的不足之处在于深度信息需要人为设定阈值将前景和背景进行划分，且当前景和背景重叠边缘的深度信息相似时，深度信息也无法很好进行前景和背景的估计。除此之外，由于彩色图像和深度图像需要分两次采集，其坐标映射需要借助于仿射变换，这会导致部分抠像误差。

2.4.2 结合 Robust Matting 的 TOF 深度抠像算法

在传统基于 RGB 的 Robust Matting（见上文 2.3.3）的基础上，Cho 等人将深度信息作为第四维信息，对于 Robust Matting 无法精确处理的前景和背景颜色相近处的透明度误差，同样地利用深度信息产生的透明度替换该区域透明度^[38]。

该算法同样通过彩色相机和深度相机分别获取彩色图像和深度图像，并利用深度信息自动生成三元图，在式 (2-11) 的基础上，对于前景颜色 F_i 和背景颜色

B_i 相似的未知像素，结合基于颜色计算得到的透明度与基于深度信息得到的透明度进行最终 α 值的求解。

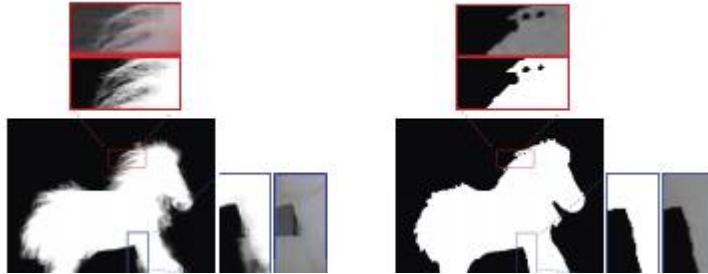


图 2-10 基于彩色图的抠像（左）与基于深度图的抠像（右）结果对比^[39]

为增大前景与背景之间的颜色差异，该算法首先用 GIE Lab 空间代替 RGB 颜色空间求取其差异值 ΔE ：

$$\Delta E = \sqrt{(L_f - L_B)^2 + (a_f - a_B)^2 + (b_f - b_B)^2} \quad (2-29)$$

基于差异值 ΔE ，定义权值 w_c 以判断该区域是否需要结合深度信息：

$$w_c = \begin{cases} 0 & \text{if } \Delta E < T \\ 1 & \text{otherwise} \end{cases}$$

其中，阈值 $T \in (6, 10)$ ，则最终的透明度计算公式：

$$\alpha = w_c \hat{\alpha}_c + (1 - w_c) \hat{\alpha}_d \quad (2-30)$$

最后，对求解得到的 α 掩膜图进行平滑处理，定义像素 i 和虚拟结点 F 和 B 的数据权值 $W_{i,F}$ 和 $W_{i,B}$ ：

$$W_{i,F} = \gamma \cdot \hat{\alpha}_i \quad (2-31)$$

$$W_{i,B} = \gamma \cdot (1 - \hat{\alpha}_i) \quad (2-32)$$

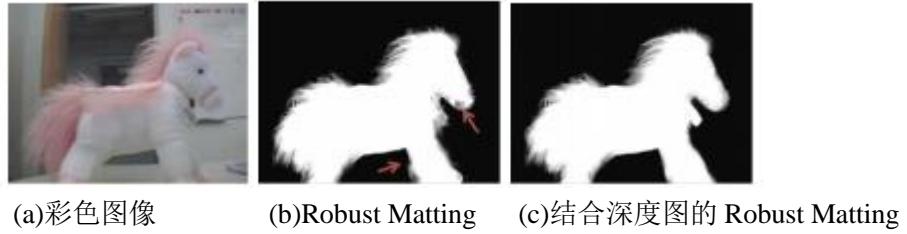
文中 $\gamma = 0.1$ 为平滑因子，其值越小，掩膜图越趋于平滑。

对于边权重，即相邻像素 i 和 j 之间的权值 $W_{i,j}$ ，也引入深度信息：

$$W_{i,j} = \sum_k^{|N_k|} \frac{1}{|k|} \left(1 + (I_i - \mu_k) \left(\sum_k + \frac{\varepsilon}{|k|} I_4 \right)^{-1} (I_j - \mu_k) \right) \quad (2-33)$$

其中， N_k 为包含像素 i 和 j 的 $m \times m$ 大小矩阵内的所有像素集合。 μ_k 为每个窗口 N_k 中向量 (r, g, b, d) 的平均向量， I_4 为 4×4 的单位矩阵。在文中设置可调参数 $m = 3$ ， $\varepsilon = 0.00001$ 。最后利用随机漫步算法进行透明度的平滑。

其算法测试结果图如下所示：

图 2-11 经典 Robust Matting 与基于 ToF 的 Robust Matting 结果抠像对比^[39]

如上图，当前景颜色和背景颜色较接近但深度上具有较强边缘时，结合深度图的 RGB-D Robust Matting 算法性能上优于基于 RGB 的传统算法。

2.5 本章小结

本章主要介绍了三种经典的深度图像获取技术，包括双目测距技术，结构光技术以及飞行时间（TOF）技术。从各技术的原理展开，阐述了三种技术的应用场景以及各自的优劣势。特别介绍了本文中使用设备 Kinect 携带的 TOF 技术。

接着介绍了抠像算法的基本原理，从数字抠像的原理方程 $C = \alpha F + (1-\alpha)B$ 展开，将自然图像的数字抠像方法从基于颜色采样的抠像方法、基于传播的抠像方法以及结合采样和传播两者优势的方法三大类展开，分别介绍了其中最经典的抠像方法：贝叶斯抠像方法、泊松抠像方法以及 Robust Matting 抠像方法。详细阐述了每个算法的基本原理及实现步骤，以及在自然图像中存在的不足。

最后介绍了基于 TOF 的 RGBD 深度抠像算法，将深度信息引入传统的抠像算法中，利用深度信息弥补传统的基于 RGB 的抠像算法中对于前景和背景重叠区域颜色相近区域的处理上的不足。同时利用深度信息自动生成三元图，从而大大减少了用户的手动交互，并能获得较好的成果。

本章的理论基础主要是为后文基于 Kinect 的抠像算法的研究打好理论基础。

第三章 基于 Kinect 的抠像算法研究

3.1 Kinect 工作原理

3.1.1 Kinect 简介

Kinect 最初是作为 XBOX360 的体感周边外设而发布的，Kinect 以其独特的即时动态骨骼追踪、语音输入控制和肢体识别技术取代了传统的手柄以及脚踏控制器，其“免控制器的游戏与娱乐互动体验”不仅为广大的游戏爱好者带来了全新的非凡游戏体验，在计算机视觉领域也引发了新的研究热潮。而微软并没有将这一先进的技术局限在游戏行业中，而是于 2011 年 6 月推出了 Kinect for Windows SDK Beta，并不断更新，从而将 Kinect 技术推广到 Windows 平台。SDK 开发包为开发者直接使用 Kinect 开发基于 Kinect 体感交互技术的各领域内应用提供了强大的支持。

2013 年 5 月 28 日的 XBOX One 发布会上，Kinect V2 作为次世代主机的体感外设被推出，研究者可以基于 Kinect 获取和识别的语音、手势、场景信息和用户体感信息进行更多的深度开发。

Kinect 作为消费级的深度相机，其独特的 RGBD 技术使得同时获取高分辨率的深度信息和彩色图图像信息成本大大降低，也使得该技术在各领域内的应用应运而生。如图 3-1 展示了 Kinect 在各领域中的研究和应用热点^[39]。Kinect 硬件以及技术被广泛地应用于物体的检测、识别和跟踪，场景识别，人体姿势和动作识别，手势分析以及室内 3D 重建等领域，在其推动这些科研领域发展的同时，科研工作者的研究进展也同样推动着 Kinect 技术不断提高和发展。

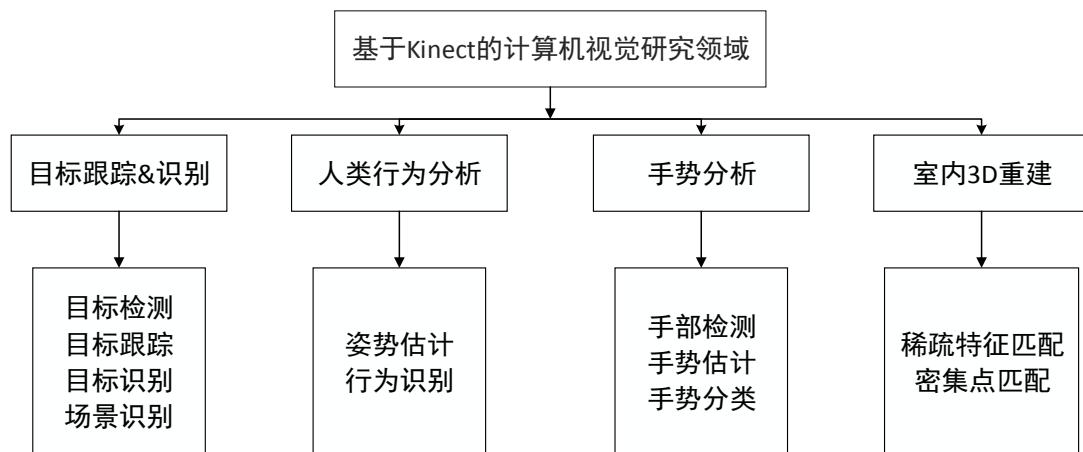


图 3-1 Kinect 研究与应用领域

Kinect 硬件设备由一个彩色摄像头，一个深度传感器以及一组多阵列麦克风

组成，因此 Kinect 可以同时获取彩色图像、深度信息和语音信息。而 Kinect SDK 包则为图像流的同步、人体的 3D 动作的捕获追踪，人脸识别以及语音识别等等技术提供的软件支持。

图 3-2 分别为一代 Kinect 和二代 Kinect，二代 Kinect 不仅改变了一代的外观，其在性能较上一代有如下几个改进：

- ① 在彩色图像和深度图像的分辨率上有了很大的提高，可提供全高清画面 (Full HD Color)
- ② 采用新的主动式红外检测，增加独立照明红外技术，可同时使用红外和彩色摄像头
- ③ 不再使用倾斜马达，在水平和垂直方向上具有更宽阔的深度和彩色视野
- ④ 改进的麦克风，采用零点平衡技术
- ⑤ 骨骼追踪的改进，可识别 6 人，每人可识别的骨骼点从 20 个增加至 25 个
- ⑥ 增加拇指追踪、手指末端追踪、打开和收缩的手势

表 3-1 展示了 Kinect V1 和 Kinect V2 传感器的配置比较：

表 3-1 Kinect v1 和 Kinect v2 的传感器配置比较

		Kinect v1	Kinect v2
颜色 (color)	分辨率	640×480	1920×1080
	帧率	30 <i>fps</i>	30/15 <i>fps</i> (根据光照)
深度 (Depth)	分辨率	320×240	512×424
	帧率	30 <i>fps</i>	30 <i>fps</i>
识别人体数量 (Player)		6 人	6 人
识别人物姿势 (Skeleton/Body)		2 人	6 人
关节数 (Joint) -		20 关节/人	25 关节/人
检测范围 (Range of Detection)		0.8-4.0m	0.5-4.5m
手的开闭状态检测 (Hand State)		借助 Developer Toolkit	可使用自带 SDK 实现
角度 (Angle) (Depth)	水平 (Horizontal)	57°	70°
	垂直 (Vertical)	43°	60°
底座马达 (Tilt Motor)		有	无 (需手动调节)
同时运行多个 APP		不能 (一次运行单个)	能

由于本文所研究的算法都在 Kinect 二代的基础上实现，因此，下文中所提到的 Kinect 均为图 3-2 中右侧所示的 Kinect 二代传感器。



(a)Kinect一代(2010年6月发布) (b) Kinect二代(2013年6月发布)

图 3-2 两代 Kinect 外形对比

1) Kinect 的硬件组成

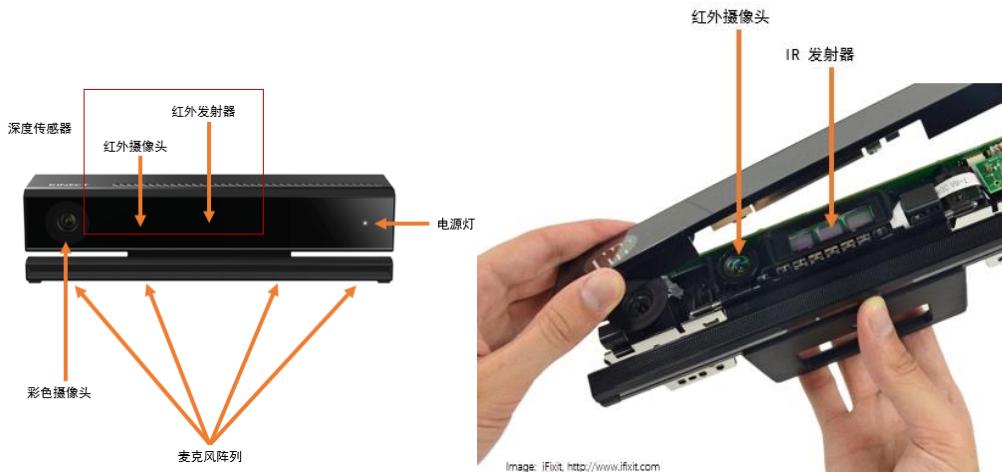


图 3-3 Kinect 硬件组成

图 3-3 展示了 Kinect V2 的硬件组成结构，其包括三个部分：彩色摄像头（RGB Camera）、深度传感器（Depth Sensor）以及多阵列麦克风（Microphone Array）。

① 彩色摄像头

彩色摄像头可传送包含 RGB 三个 8 比特通道的视频流。Kinect 提供了 1920×1080 的高分辨率彩色图像，帧率根据设备所在的光照条件可达到 30 fps 或者 15 fps 。

② 深度传感器

深度传感器由一个红外线发射器（IR Emitters）和一个红外摄像头（IR Camera）组成。深度信息即 Kinect 成像范围内的对象与 Kinect 所在平面的 3D 空间距离。在深度传感器中，先由红外发射器发射各个方向的红外线，通过红外摄像头根据所接收的红外信息获取红外图像，而深度图像经由红外图像计算得到。深度传感器可获取在 0.5 到 8 米之间的对象的距离信息（单位为米），从而生成分辨率为 512×424 ，帧率为 30 fps 的 16 比特的深度图像。

③ 麦克风阵列

Kinect 的麦克风阵列由 4 个麦克风组成，其提供了可上下调节的底座，并采

用了零点平衡技术。Kinect 的语音识别技术是一种基于距离的识别技术，其根据四个麦克风阵列同时采集音频，消除噪声，并进行语音识别从而定位定位，同时采用零点平衡增加了语音识别的精度和稳定性。

2) Kinect 的数据流

借助独特的主动式红外检测技术，高精度的摄像头和麦克风阵列，Kinect 可同时获取包括彩色图像，红外图像，深度图像，人物索引以及人物骨骼和语音流在内的 6 中数据流，如图 3-4 所示。表 3-1 中总结了各数据流的来源，图像分辨率和获取帧率等信息。



图 3-4 Kinect 的 6 种数据流[来源：Kinect 官网]

表 3-2 Kinect 数据流参数分析和说明

Kinect 的数据流				
名称	来源	分辨率	帧率	说明
彩色图像 ColorFrameSource	彩色 摄像头	1920×1080	30 <i>fps</i> 光照好 15 <i>fps</i> 光照弱	展示场景中彩色图像，支持多种格式
红外图像 InfraredFrameSource	红外 摄像头	512×424	30 <i>fps</i>	去掉光照的场景信息 人脸识别等需要光照一致性的应用
深度图像 DepthFrameSource	深度 传感器	512×424	30 <i>fps</i>	对象 距离 Kinect 所在平面的距离
人物索引 BodyIndexFrameSource	深度 传感器	/	/	根据前后帧信息对比检测人体，可用于抠像技术
人物骨骼 BodyFrameSource	深度 传感器	/	/	三维人体信息，可用于建模，肢体识别
音频流 Audio Source	麦克风 阵列	/	/	高指向性扬声器，可焦点追踪

3) Kinect 多层次的架构设计

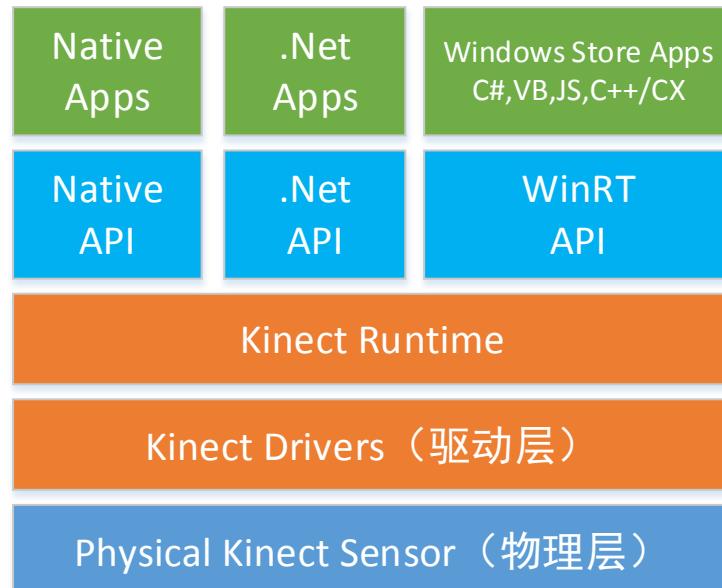


图 3-5 Kinect 的多层次架构设计

图 3-5 展示了 Kinect 的架构层次，从底层往上依次为物理硬件层，驱动层，接口层和应用层。Kinect 的 API 支持开发者使用 C#，VB，JS，C++ 等多种语言进行应用层上不同平台应用的开发，而 Kinect 强大精确的人脸识别、骨骼识别和跟踪技术也为基于 Kinect 的开发提供强大的技术基础。同时一台 Kinect 设备可同时运行多个基于 Kinect 的应用，这意味着研究者可以在运行应用的同时实时对数据进行读取和分析，极大的增强了应用开发的透明度和可控制性。

3.1.2 Kinect 深度图

Kinect 的光学成像采用主动测距技术，其深度信息主要由深度传感器获取得到，Kinect V2 用飞行时间技术（Time of Flight）取代了一代的结构光编码技术，由红外线发射器主动发射连续红外短脉冲，并且在发射处接受目标物的反射光，由红外摄像头分析读取到的不同物体反射信息的往返时长（即飞行时间）来判断物体距离 Kinect 的距离，从而形成深度图像。

1) Kinect 深度测量原理

Kinect 传感器中，基于 TOF 的深度测量方法依赖于一个快速时钟信号，该时钟信号由三条激光二极管组成频闪发射器，同时从扩散器发射短脉冲红外光对场景进行检测。激光脉冲检测场景中的物体后再返回 Kinect 设备的红外摄像头进行进一步分析。而 Kinect 深度传感器的独特之处在于，它将每个像素分为两个累加器（accumulator），并由时钟信号控制当前活动的累加器对返回的红外脉冲进行寄存。在这一周期中，当脉冲光返回摄像头，由时钟信号转换当前寄存入射光的累加器，先结束当前分配累加器并开始二号累加器的工作。通过比较每个累加器获取的光脉冲的数量或是比例即可计算距离：相比于一号累加器，二号累加器获取的反射脉冲光越多，则这部分场景距离传感器的距离越远。原因是随着周期的进行，需要寄存的返回距离增多，而传感器需要通过调制时钟频率来消除多个距离之间的干扰。

2) Kinect 深度传感器测量误差

本小节主要讨论 Kinect 在使用 TOF 技术进行深度测量时存在的误差，从而分析 Kinect 深度系统的测量精度。根据 2.1.3 节中所描述的 TOF 工作原理，TOF 作为一种主动式的红外深度信息测量技术，其发射的主动式信号是经过幅度调制的人眼不可见的红外脉冲信号，并且其发射功率范围对于人眼较为安全。

Kinect 深度信息的测量误差通常由于传感器本身、外界测量条件以及被测物体的表面条件三种情况产生。本文中将外界测量条件以及被测物体表面导致的误差都归为外界动态误差，下文将从传感器误差与外界动态误差两个方面对 Kinect

深度测量的误差进行分析。

① 传感器误差

传感器误差也称为“系统误差”、“摄像机静态误差”⁴⁰，主要来源于 Kinect 传感器本身的成像系统，其会影响单个像素点的精度。比如深度传感器上存在的“被摄对象边缘叠加误差”和“镜头弯曲误差”等。这类误差出现概率较高，表现形式稳定，因此可以用固定的修复算法进行去噪补偿。

传感器误差的来源之一是 Kinect 中的信号发射阵列。如上文所述，传感器中多个 LED 二极管发射出独立的红外脉冲，这会形成相互间的干扰，因此 Kalman 等人提出的误差校正模型就可以应对这类误差。

深度传感器噪声也会引起系统误差，常见的有低照明环境中的成像噪声、特殊场景中产生的量化噪声和光电噪声等，同样的，这类噪声也可以较为轻松地被抑制。

除此之外，持续测量中的曝光误差，目标边缘误差也是不容忽视的。曝光时间越长，深度信息的测量误差率越高，其数据越不可靠。实验证明在进行室内深度测量时，曝光时间为 1000us 左右时其测量结果更精确。在测量过程中，部分测量信号在返回 Kinect 传感器表面时被反射或漫射，从而又一次进入测量周期，直到最终被原像素或附近像素接收^[41]。这些经过多次测量的信号相位与正常的测量信号相位进行叠加，从而产生了导致了测量距离的目标边缘误差。这种误差会在测量目标前景与背景之间产生冗余的数据，既不属于前景也不属于背景，而当被测对象表面较为光滑时，这类误差并不明显，因此通常采用平滑技术或是边缘标记来处理这一误差。

② 外界动态误差

在 Kinect 的深度测量过程中，外界测量条件，如光照条件和测量角度的复杂多变等同样会引起一定的测量误差，由于这类误差具有随机性和变化性，因此很难用一个数学统计模型对其进行定量分析。因此目前并没有通用的算法对其进行处理。

光照条件会影响视觉测量的精度^[42]。由于 Kinect 中的 TOF 技术以红外线为光源，因此当场景中存在如太阳光等的红外光源时，会对测量精度造成较大的影响。同样的，当场景中存在多个 Kinect 同时进行测量时，也会对彼此的测量结果造成干扰。

测量角度引起的误差主要由测量过程中测量信号未始终保持与测量目标垂直引起的。当测量两个成角度的平面时，Kinect 发出的测量信号被“聚集”在该角度所成的角落，导致大量信号无法发散而产生多次反射后返回传感器，这些多余的信号和正常的测量信号发生叠加从而产生了大量的误差。

在不同场景进行深度测量时，场景的几何特性，场景中目标对象的表面材质和颜色特性也会引起测量误差。

首先场景的几何特性，包括场景中目标对象距离传感器的距离会影响 Kinect 的深度测量精度。Kinect 的有效测量距离为 0.5-4.5 米，其传感器的探测效果随距离变化如下图 3-6 所示：

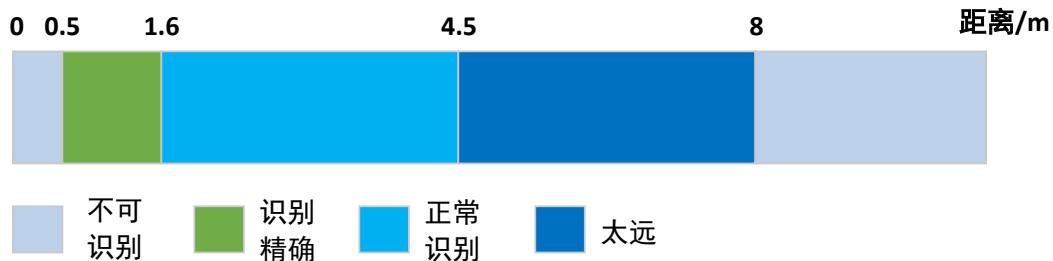


图 3-6 Kinect 探测距离示意图

Thomas 等人通过实验分析了 Kinect 二代的深度测量精度与测量距离之间的关系^[43]。实验人员将 Kinect 安装在一辆滚动的车上，使其垂直指向一面平坦的白墙。Kinect 设备从工作范围的最小距离，即 0.5 米处开始，每次移动几厘米一直到最大的测量距离——4.5 米。每一次停留测量时，深度图像中心 100×100 的子块中的深度值会被记录超过 250 帧，从而根据整个子块的均值计算每个像素深度值的统计分布。

如图 3-7 展示了本次实验结果。当距离从 0.5 米到 1.6 米变化时，深度测量的标准误差都保持很小 ($< 1.5\text{mm}$)，并且其变化是随机的。当距离超过 1.6 米一直到 4.5 米，标准误差随着距离的变大呈现出线性增长。而最远距离点的平均标准差也低于 3.5mm，这意味着 Kinect 二代的深度精确值相比一代已经上了一个数量级^[44]。

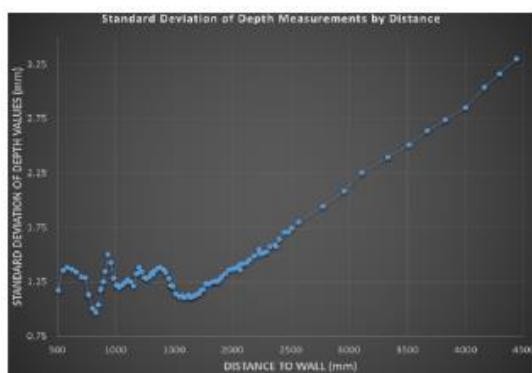


图 3-7 平坦表面上深度测量随距离变化的标准差趋势^[44]

另一方面，当场景中的前景对象对其身后区域形成遮挡，则会导致 Kinect 发射出的红外光被该物体遮挡而使部分原本应该接收红外光并发生反射的区域无法返回飞行时间数据，也就无法测量其深度信息。如图 3-8 展示了当目标对象部分被遮挡时产生的误差原理图。

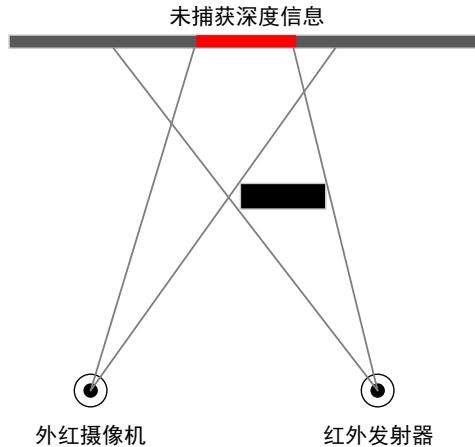


图 3-8 黑洞生成的基本原因

最后场景中目标对象的表面材质也是造成测量误差的一个因素。当物体表面过于光滑时，红外信号在该表面会产生镜面反射，导致传感器无法接受到由该段表面返回的信号，从而无法获取该物体的深度信息。因此在深度测量中，镜子和玻璃材质物体在深度图中的测量值通常为 0。除光滑平面外，当目标对象表面是吸光材质（如黑色不光滑表面）时，经由目标对象的红外光会被对象吸收而非反射，导致传感器无法捕获反射光。同样的，这一部分在深度图中的值通常为 0。如图 3-9 所示：

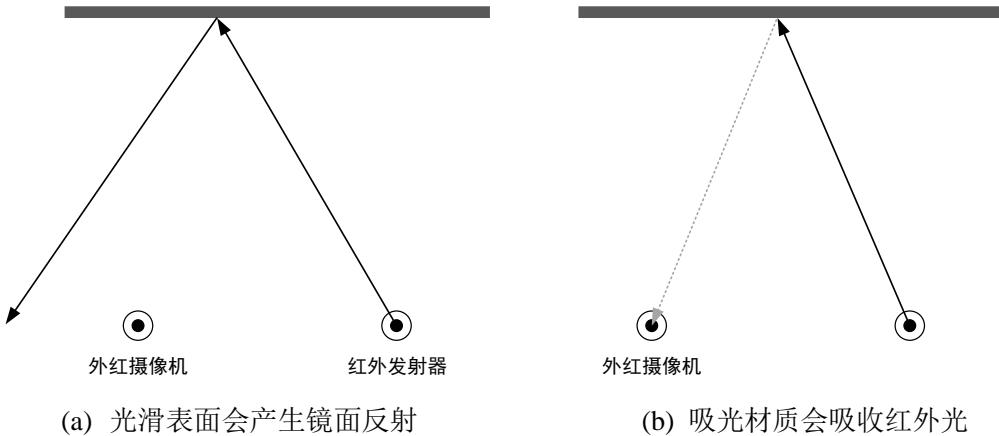


图 3-9 黑洞生成的其他原因

深度测量结果中，测量误差最典型的表现为“空洞”，即深度值为 0 的像素点在深度图像中表现为黑色区域。

如图 3-10 展示了 Kinect 获取的彩色图及其对应的深度图。从中可以观察到本节上述的 Kinect 测量误差信息，包括随机噪声，边缘误差以及“空洞”像素等。



(a)Kinect 彩色图

(b)Kinect 深度图

图 3-10 Kinect 获取的彩色图像以及对应的深度信息图

上图测试在室内日光灯下进行，图 b 中，对应左侧的彩色图，可以清晰的分辨深度图中的测量误差部分：①深度图的四周出现了大量的随机噪声，即上文所说由传感器误差引起的噪声，即“闪烁”；②由前后景之间物体遮挡引起的人物及其他物体轮廓边缘误差，这部分误差也被称为“飞行像素”；③电视机轮廓为镜面光滑材质、电脑主机以及右侧椅子为吸光黑色材质，因此这几个区域在深度图中都表现为“空洞”区域。

本节主要叙述了 Kinect 进行深度测量的原理，并分析了 Kinect 获取的深度信息的误差种类，成因以及处理方法。为下文中利用深度信息进行抠像提供理论基础。

3.1.3 深度图与彩色图的坐标映射

由 KinectV2 彩色摄像头获取的彩色图的标准分辨率为 1920×1080 ，其拍摄视角为 $84.1^\circ (h) \times 53^\circ (v)$ ，深度图的分辨率为 512×424 ，其拍摄视角为 $70.6^\circ (h) \times 60.0^\circ (v)$ ，因此这两种图像在中间会有部分重叠，但无法相互覆盖。在进行算法处理前，首先需要对 Kinect 的彩色图与深度图进行坐标映射。Kinect 的彩色高清图像摄取的场景在宽度上大于深度图，而在高度上小于深度，这种不完全重叠在获取点群时具有现实意义：当捕获与 RGB 色彩信号相关的点时，点群可直接进行垂直方向上的裁剪。在 Kinect 的坐标映射中，采用的坐标系如下图 3-11 所示。

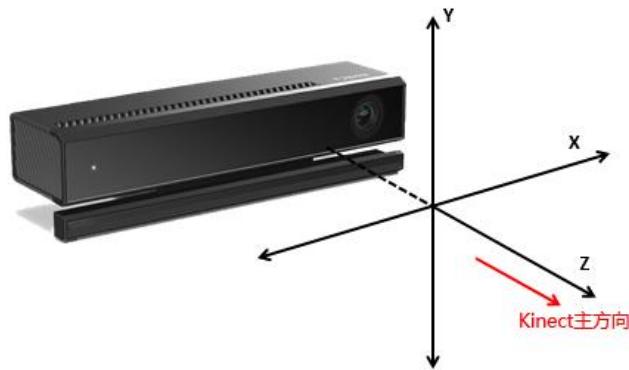


图 3-11 Kinect 的主坐标系设置

Kinect 中提供了一套应用于不同图像空间的坐标系统，如表 3-3 所示：

表 3-3 Kinect 的空间坐标系统

坐标系统	应用图像	维度	单位	分辨率	初始点
ColorSpacePoint	彩色图像	2	像素	1920×1080	左上角
DepthSpacePoint	深度图像 红外图像 人物索引	2	像素	512×424	左上角
CameraSpacePoint	人物骨骼	3	米	/	红外摄像头

在对彩色图和深度图进行基本的坐标映射和裁剪之后，两者之间仍存在着边缘的不对齐，如图 3-12 (a) 所示，红色线条为利用 canny 算子检测到的彩色图像的边缘信息，在 3-12 (b) 的放大图中可以看到在人物裤腿处存在着明显的边缘不对齐。这一误差是由于色彩摄像头和红外摄像头的光学畸变引起的[11]。

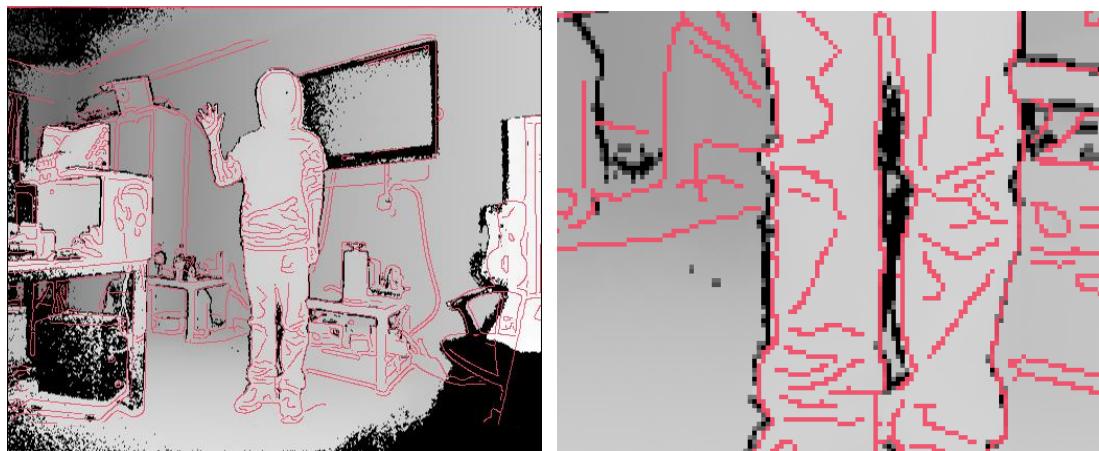
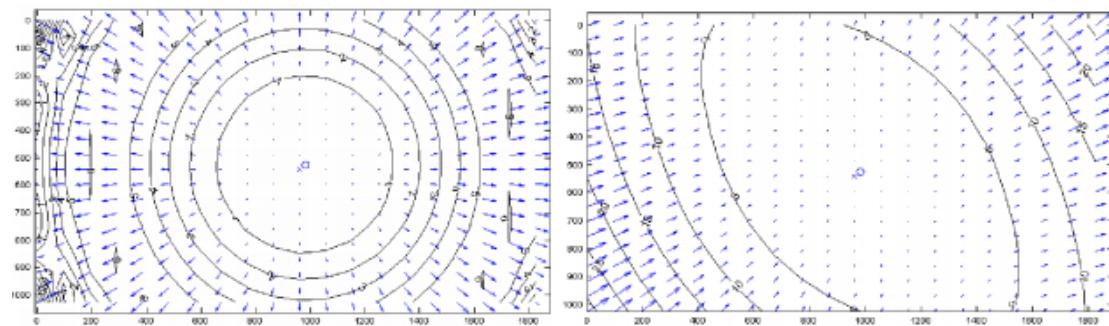


图 3-12 Kinect 深度图中的边缘不对齐

文献[43]中, Thomas 等人对 Kinect 的彩色摄像头和红外摄像头进行了标准相机标定, 实验人员采用一张油墨印刷的棋盘, 从两个摄像头中分别获取 20 张棋盘不同角度和位置的图像。如图 3-12 所示, 接着将图像导入 Bouguer 的相机标定工具中计算固有参数。图 3-14 和图 3-15 分别展示了 Kinect 彩色摄像头和红外摄像头的径向畸变和切向畸变。



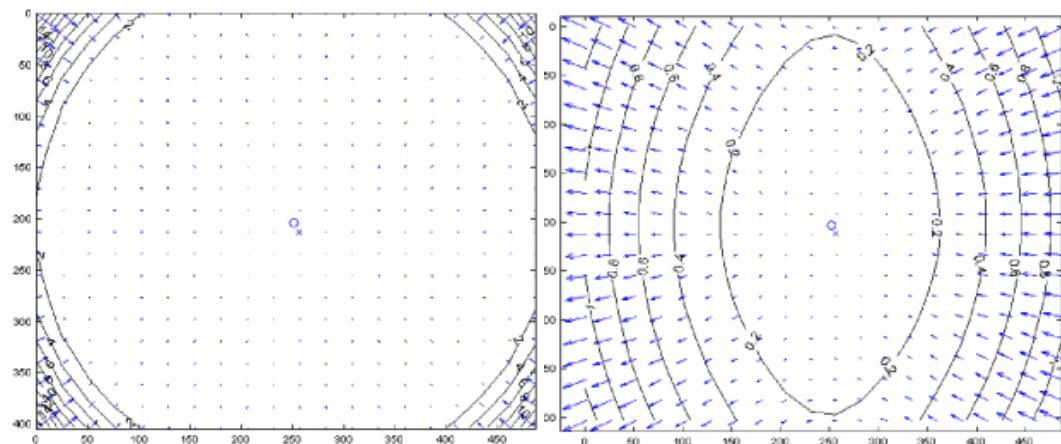
图 3-13 Thomas 实验中彩色摄像头和红外摄像头的标准采样图^[43]



(a) 彩色摄像头的径向畸变

(b) 彩色摄像头的切向畸变

图 3-14 彩色摄像头的径向畸变 (左) 和切向畸变 (右)^[43]



(a) 红外摄像头的径向畸变

(b) 红外摄像头的切向畸变

图 3-15 红外摄像头的径向畸变 (左) 和切向畸变 (右)^[43]

径向畸变是指图像像素点从真实位置 (理想位置) 指向图像中心或是远离图像中心的位移, 图中的环线上的数字即表示径向畸变和切向畸变量, 单位为毫米。

对比图 3-14 (a) 和 3-15 (a) 可知, 彩色摄像头的径向畸变大于红外摄像头, 这很大程度上导致了彩色图边缘与对应的深度图不对齐。而对于大多数研究者来说, 工厂设置的标定值已经能够满足实验需要, 而微软在推出的 Window v2 SDK for Kinect 中也提供了内置的坐标映射方法, 可快速地将红外图像以及深度图像的像素点映射到彩色图中, 且这一坐标映射可实现深度图像到世界坐标系中 3D 坐标之间的实时转换。

本节首先介绍了 Kinect 的发展过程及设备特色, 同时分析了 Kinect 深度图像测量的原理及其测量过程中产生的两类误差, 最后就 Kinect 中彩色图和深度图之间的坐标映射原理和方法进行了分析和总结。

3.2 Kinect 深度图的平滑和滤波

由于 Kinect 深度图存在由传感器和外界测量条件等引起的测量误差, 因此需要借助算法对深度图进行修复和平滑操作。本文中针对 Kinect 深度图中“空洞”的形成特点, 以及 Kinect 特有的彩色图和深度图之间的对应关系, 提出将引导滤波引入基于 Kinect 的抠像算法的深度图预处理中。引导滤波^[45]是 Kaiming He 等人提出的一种局部线性移可变的图像滤波器, 其具有良好的保边去噪性能 (即在对平滑图像消除噪声的同时还能保持良好的边缘特性), 因此引导滤波提出后就被广泛地应用于图像平滑去噪、超分辨率复原处理和数字图像增强等领域。

不同于传统的基于 RGB 空间的引导滤波, 本文将 Kinect 深度图像引入引导滤波中, 结合彩色图像进行处理, 从而得到更精确更接近人眼的视觉效果。在此基础上, 本文巧妙地运用引导滤波的迭代使用来增强滤波的精度和效果, 从而得到更符合视觉效果的滤波结果。

3.2.1 引导滤波的基本原理

引导滤波是一种基于局部线性模型的滤波过程, 该模型认为, 函数上某一点与其邻近区域的点存在线性关系, 因此一个复杂的函数可以视为多个局部线性函数的组合, 当计算函数上某一点的值时, 只需计算包含该点的所有线性函数的值并进行平均。

如图 3-16 展示了此线性模型:

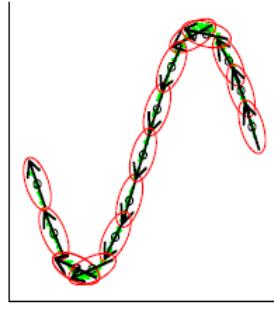


图 3-16 引导滤波的局部线性模型

引导滤波器通过一幅引导图像 I 对输入的待处理图像 p 进行滤波处理，最终输出的图像 q 既能保留输入图像 p 的整体特征，同时能充分获取引导图像 I 的边缘梯度特征。则以像素 k 为中心的窗口 w_k 中引导图像和输出图像之间的线性关系如下：

$$q_i = a_k I_i + b_k, \forall i \in w_k \quad (3-1)$$

上式中， w_k 为半径为 r 的方形窗口， a_k 和 b_k 为窗口中的线性系数。其中引导图像 I 既可以是特定应用场景中设定的外部图片，也可以直接取输入图像 p 。

对式(3-1)两边取梯度：

$$\nabla q = a \nabla I \quad (3-2)$$

即输出图像 q 与输入图像 I 具有相似的梯度和边缘一致性，这也解释了引导滤波保边去噪的特性。引导滤波的关键步骤在于求得式 (3-1) 中线性系数 a_k 和 b_k 的最优解，即线性回归，以达到线性拟合中输出图像 q 和输入图像 p 之间的差异最小，则将问题转化为求以下目标函数的最优化求解：

$$E(a_k, b_k) = \sum_{i \in w_k} ((a_k I_i + b_k - p_i)^2 + \varepsilon a_k^2) \quad (3-3)$$

式中， ε 是调节引导滤波效果的正则参数，防止 a_k^2 过大。通过最小二乘法可以得到 a_k 和 b_k 的最优解如下：

$$a_k = \frac{\frac{1}{|w|} \sum_{i \in w_k} I_i p_i - \mu_k \bar{p}_k}{\sigma_k^2 + \varepsilon} \quad (3-4)$$

$$b_k = \bar{p}_k - a_k \mu_k \quad (3-5)$$

其中， μ_k 为 I 在窗口 w_k 中的平均值， σ_k 为其标准差， $|w|$ 是窗口 w_k 中像素的数量， \bar{p}_k 为待滤波图像 p 在窗口 w_k 中的均值。

对于单幅图像，每个像素都会被多个窗口包含，因此该像素可由每个窗口中

的线性函数计算出不同的 q_i 值。因此，求某一像素的输出值，只需取包含该点的所有线性函数值的平均，则在计算出所有窗口系数 a_k 和 b_k 后，滤波输出图像为：

$$q_i = \frac{1}{|w|} \sum_{i \in w_k} a_k I_i + b_k = \bar{a}_i I_i + \bar{b}_i \quad (3-6)$$

其中 w_k 是所有包含像素 i 的窗口， k 是其中心位置。

当把引导滤波用作边缘保持滤波器时，通常设置 $I = p$ ，若 $e = 0$ ，显然 $E(a, b)$ 的最小值的解为 $(a=1, b=0)$ ，从上式可以看出，此时引导滤波没有起任何作用，输出图像 q 就是输入图像 I 本身。若 $e < 0$ ，在像素无明显变化的区域，有 a 趋近于 0，而 b 近似于 \bar{p}_k ，即进行了加权均值滤波；而在像素变化明显的图像区域如边缘， a 约等于 1， b 约等于 0，对图像没有滤波效果，即保持了图像原有边缘。而 e 为变化大小的界定参数，若窗口大小固定， e 越大，滤波效果越明显。

引导滤波结果如下图 3-17，图(a)为待滤波原图，(b)和(c)为计算得到的 a_k 和 b_k ，(d)为最后得到滤波图。从图中看出，在变化较剧烈的部分，如边缘等， a 的值接近于 1，显示为白色， b 的值接近于 0，显示为黑色，而在变化平坦的区域 a 的值接近于 0， b 的值为平坦区域像素的均值，相当于进行了平滑的作用，这与上述引导滤波的规律相符。从最后得到滤波效果也可以看出，引导滤波在对图像中间像素平坦的部分进行平滑的同时，保留了清晰的边缘，对于较细节的毛发也有很好的保留作用。

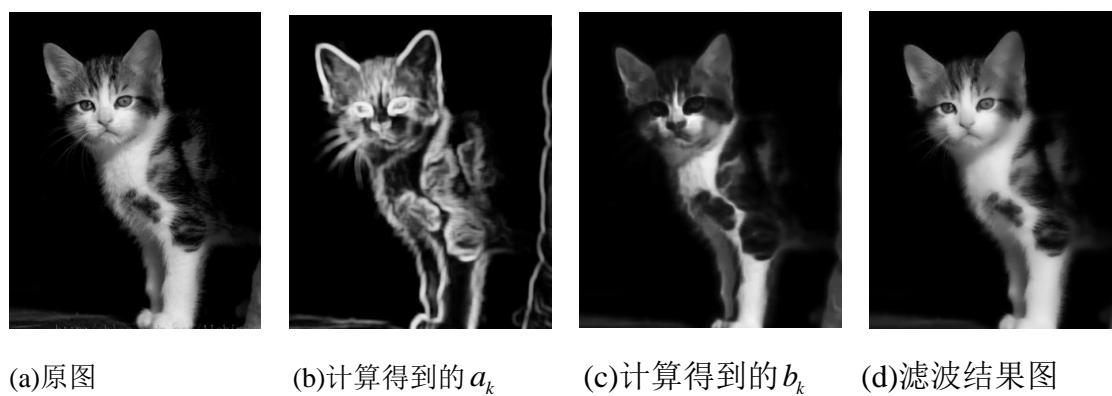


图 3-17 引导滤波器滤波过程图及效果图

(滤波窗口半径为 8, e 的值为 50)

3.2.2 基于深度图像的引导滤波

综合上述引导滤波器良好的保边去噪特性，以及 Kinect 特有的彩色图与一一对应的深度图，本文将引导滤波引入到了基于 Kinect 的抠像算法中。将 Kinect 获取的彩色图像作为引导图像 I ，利用对应的深度图中提取出的前景对象的二值

图作为输入图像 p 。

图 3-18 为 Kinect 获取的深度图以及对应的人体前景分割图（人物索引图）。

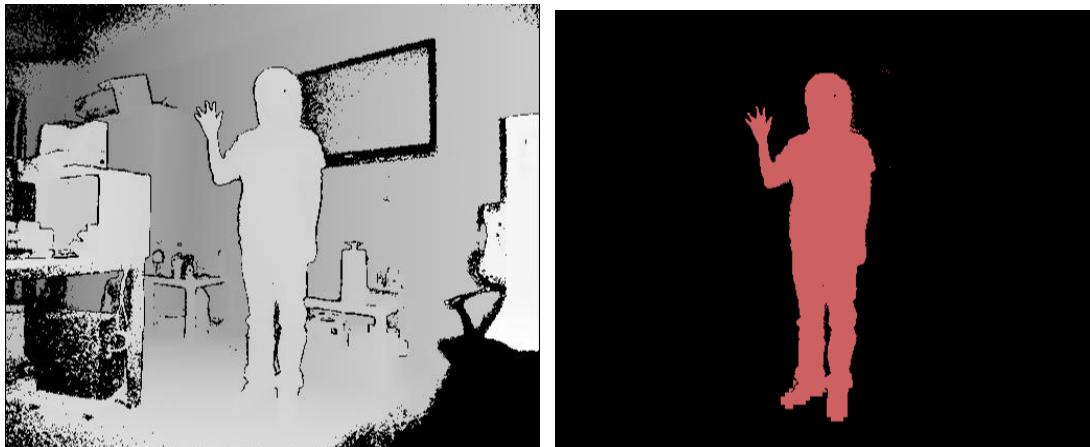


图 3-18 Kinect 获取的深度图与对应的前景分割图

传统图像处理中前景对象的获取主要有阈值法，基于边缘的方法和基于区域的方法。其中最常见的是阈值法，只需将一个亮度常量作为阈值即可区分图像中的前景和背景，该方法计算简单但不适用于背景较为复杂的图像；基于边缘的分割法中，用户可手动绘制前景对象的边缘曲线并进行分段优化，从而提取需要前景的边缘，这增加了人工成本，且其对噪声抑制能力较差；基于区域的方法则依据纹理及像素的统计特性进行提取，该方法能有效抑制噪声，但容易出现过度分割，从而产生大量零碎区域，如比较经典的区域生长法等。

由于深度图像具有特色的深度区域信息显示，大多数情况下，当目标前景为物体时，只需使用阈值法即可将前景较好的分割出来，得到较粗略的前景目标对象。从 3-18(a)可以观察到，Kinect 深度图中存在一个比较明显的问题，即前景目标为人体时，由于人体脚部与地面接触的位置深度信息相似，因此在深度中并没有区分出两者。而 Kinect 也提供了这一问题的解决方案，即它强大的人体识别技术，可获取人体数据，即 3.1 章节中提到的人物索引数据（BodyIndex），其优势在于其不需要人为设定阈值或是平面来区分前景和背景，并且对于深度信息无法区分的鞋底区域也能较准确地识别出人体所在的区域。同时在 Kinect 获取的人物索引数据中，我们也可以清楚的看到三个缺陷：①人物边缘伴有部分随机噪声②人物中间会产生一些“空洞”，特别是当人物戴有黑色镜框时③边缘锯齿化如图 3-18(b)中所示。

如图 3-19 为对传统的物体粗略分割图进行引导滤波的实验结果图。左图将一个毛绒玩具的彩色图像作为滤波器的引导图像，中图为待处理的输入图像为对应的粗略的前景分割图，右图为原图利用彩色图进行引导滤波之后的输出结果。其中参数设置为 $r = 60$, $\varepsilon = 10^{-6}$ 。



(a)引导图像-彩色图 (b)待处理图像-阈值分割图 (c)引导结果

图 3-19 对物体的粗略前景分割图进行引导滤波

由上图中的实验结果可以发现，滤波结果图很好的保留了引导图像 3-19(a)中边缘特性，如周围精细的绒毛，而同时也很容易发现其存在的不足，即在前景中心的区域，也保留了引导滤波的一些梯度细节。通过下一个实验中可以看出不同参数调节对边缘和边缘内部细节的处理效果的影响。

图 3-20 为对 Kinect 中获取的深度图进行引导滤波的效果图，根据 Kinect 彩色高清图与深度图的对应关系，本实验中的引导图像为与深度图进行坐标映射之后与深度图尺寸相同的彩色图像。在对基于深度图的引导滤波进行不同的参数设置进行实验时，可以较直观的看到参数设置对于深度图像的滤波效果。

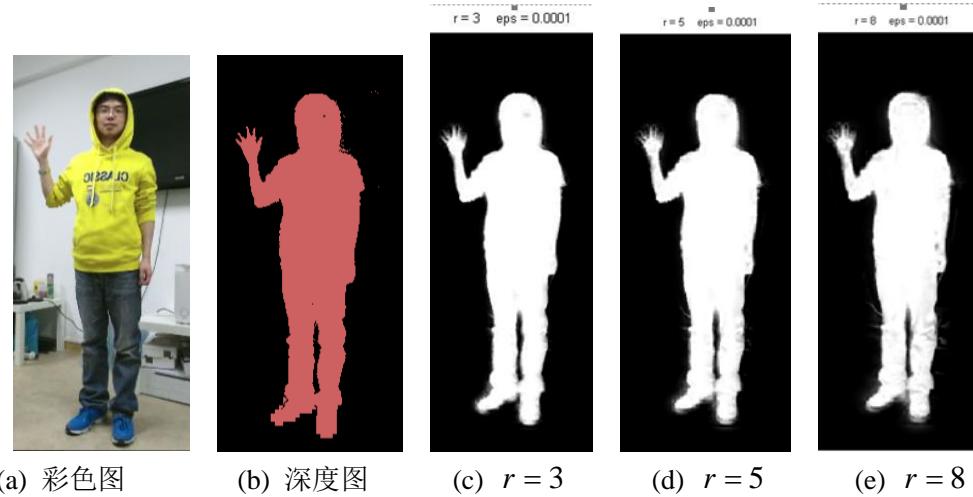
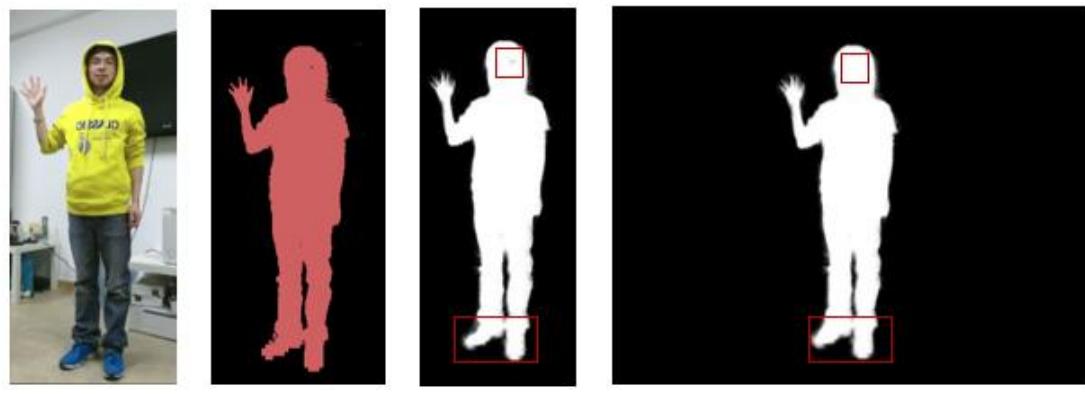


图 3-20 参数设置对基于深度图的引导滤波算法的影响

从上图可知，窗口半径 r 越小，其滤波精度越大，而对于深度图像来说，其在手臂和鞋子等边缘区域会更符合彩色图像平滑的边缘，且其在手臂与腰之间的空隙部分的处理更符合实际情况，然而同样的，高精度滤波后会使深度图像带有更多彩色图像的梯度信息和纹理信息，如帽子部分，裤子褶皱以及手掌等，这在本文中是不需要的。因此，本文选取了一组能兼顾边缘与纹理一致性的参数来进行处理： $r = 3, \epsilon = 0.0001$ 。

3.2.3 引导滤波的迭代使用

引导滤波很好地抑制了原始图像，即灰度图的大部分噪声并良好地保留了引导图像，即彩色图的边缘信息，然而在进行引导滤波的过程中，在对原始图片进行滤波时一些空洞较大的区域无法通过一次引导滤波彻底修复，且在图像增强过程中会存在某些区域的平滑效果还不够平滑。如图 3-21(c)中，面部的空洞以及脚步的残缺并没有得到很好的修复。因此综合考虑图像细节的保留和平滑效果，本文将引导滤波迭代使用，而在传统的引导滤波中，适当地增加引导滤波的次数，能增加滤波结果图的细节。而本文的滤波结果是为使原始灰度图保留尽可能精确的彩色图的边缘信息，并且对灰度图中由灰度测量误差引起的“空洞”进行修复。经过实验，本文对基于 Kinect 输出图像的引导滤波进行了 3 次迭代实验，其结果如下图 3-21 所示：



(a) 彩色图 (b) 深度图 (c) 1 次引导滤波结果 (d) 3 次引导滤波迭代结果

图 3-21 基于深度图的引导滤波的迭代使用

由上图可知，经过引导滤波迭代优化的深度信息图边缘的平滑程度基本满足预处理要求，为后文中抠像算法的使用提供更精确的输入图像。

本节主要介绍了引导滤波的原理，并提出基于深度图的引导滤波，结合 Kinect 获取的彩色图和深度图的特点，将引导滤波进行迭代优化，最终能够对 Kinect 深度图的随机噪声进行很好的抑制，同时对于基本的“空洞”进行修复，而且对于锯齿化的边缘有很好的平滑作用。经过引导滤波之后的深度图，其人物前景轮廓完整，已经初步能够达到抠像效果。为达到更精细的边缘处理效果，本文在此基础对经过引导滤波的 Kinect 深度图进行了 Shared Matting 抠像。

3.3 基于 Kinect 深度的三元图自动生成

3.3.1 数字抠像中的三元图

在很多数字抠像算法中，都需要提供三元图（Trimap）作为输入来进行进一

步的抠像计算。数字抠像中的三元图通常是一幅与原图像大小相等的图像，包含了三种信息的图像：已知背景，已知前景，以及未知区域。通常在三元图中为灰度图，背景区域和前景区域为已知区域，分别标记灰度为 0 和 255（即全黑和全白），未知区域可标记为(0, 255)之间的任一灰度。如图 3-22 所示：



图 3-22 两幅彩色图像以及对应的三元图

如上文所述：在数字抠像算法中，抠像通常是基于分割的二值图像进行处理的，因此三元图通常是对原图像的粗略划分，其中已经确定的前景和背景不需要再进行处理，只需对图像显示为灰色的位置区域进行进一步计算，来判断这一部分的像素是属于前景还是背景，或是前景和背景的线性组合，即

$$C_i = \alpha_i F_i + (1 - \alpha_i) B_i, \text{ 其中 } \alpha_i \text{ 为 } [0, 1] \text{ 之间的任意值。}$$

在传统的数字抠像算法中，三元图通常需要人工标记和分割，如图 3-22(d) 中的三元图分割需要耗费大量的人力成本和时间成本，且对于一些形状复杂的图像，其创建过程相当困难，并且使得抠像算法很难应用于实时的视频抠像算法中。而针对本文中基于 Kinect 的抠像算法的研究，为减少人工交互，同时使得算法可以应用于视频抠像中，实现了一种三元图的自动生成方法。

3.3.2 三元图的自动生成

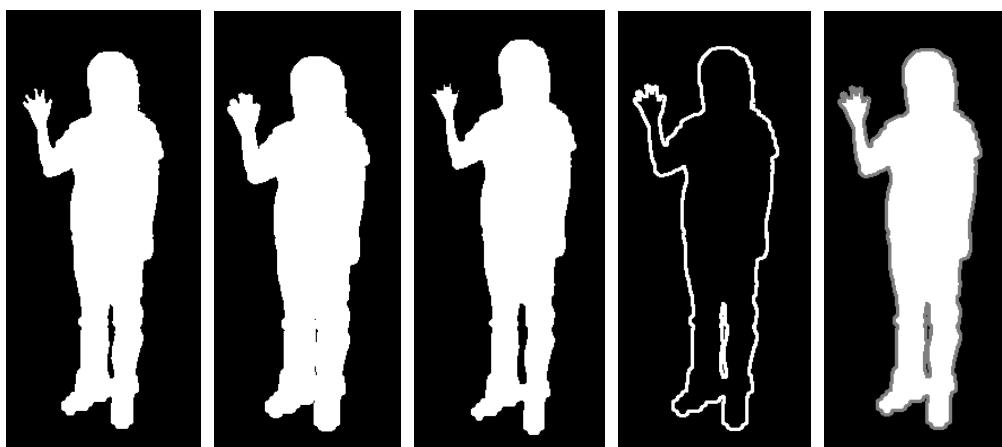
本文中，基于 Kinect 的抠像算法抠取的对象主要为人体，因此其边缘相对规则，且不存在特别复杂的毛发等区域，因此本文利用图像的形态学运算方法对引导滤波之后的深度图进行运算，从而生成包含深度图信息的三元图。

形态学运算^[46]由一系列代数运算子构成，可对二值图像进行边界提取、骨架提取、孔洞填充、角点提取和图像重建等图像处理，结合二值图像和灰度图像的特点，其基本运算包括腐蚀膨胀，开闭运算，击中击不中变换和骨架化等等。形态学运算中，其基本运算结构，即结构元素（structure element）在待处理图像中不断移动，检测图像各个部分之间的相互关系，对图像不同区域的像素信息进行采集并分析图像的结构特点，从而根据具体的应用场景，在每个像素的位置进行特定的逻辑运算。结构元素的尺寸、形状及逻辑运算种类决定了形态学运算的效果。随着图像处理领域的不断发展和探索，形态学运算在多个领域内都得到了广泛的应用，包括医学图像分析，计算机视觉，模式识别和地形监测等行业领域。

在本文的三元图自动生成方法中，采用腐蚀和膨胀操作对通过引导滤波之后对深度图进行运算，通过以下四个步骤即可得到三元图：

- ① 膨胀，得到背景图 PB
- ② 腐蚀，得到前景图 PF
- ③ 膨胀图像 PB 与腐蚀图像 PF 相减，即可得到三元图中的未知区域，标记为灰色
- ④ 三中所得的未知区域与腐蚀所得的 PF 相加，即可得到三元图。

下图展示了三元图的生成过程：



(a)原始深度图 (b)膨胀结果 PB (c)腐蚀结果 PF (d)未知区域 (e)最终三元图

图 3-23 三元图生成过程：原图 膨胀 腐蚀 未知区域 三元图

3.4 基于 Shared matting 的抠像算法

本文第二章介绍了几种经典的抠像算法，通过分析和总结可以发现，经典的数字抠像算法通常需要对三元图中未知区域内的每个像素点求解，计算量巨大；且其 α 值的计算独立于前景 PF 和背景 PB 进行，在抠像过程中需要对前景和背景进行重建。因此，即使优势最大的 Robust Matting 的抠像效率依旧比较低下，需要数秒到数分钟到时间来完成抠像的计算过程。

Shared Matting 是 Eduardo S 等人于 2010 年提出的一个可达到实时抠像速率的自然图像抠像算法^[18]。Shared Matting 的关键思想在于相邻像素点之间往往具有非常相似，甚至相同点前景背景样本对以及 α 值，因此这些小邻域内的未知像素可以通过共享样本点的方式来减少计算量。传统的基于颜色采样的方法只考虑了相邻像素之间属性相似的统计规律，但其计算过程未考虑这部分像素点本身的高度相似性，而对每一个点进行计算，导致了大量重复计算，因此，样本点之间共享样本点大大的降低了 Shared Matting 的计算量，提高了其算法的运行效率。

与此同时，Shared matting 算法将空间、光照以及概率目标函数综合作为每

组相似像素的最优前景/背景样本点对的选择依据。得到最优样本点对之后，Shared Matting 对其进行透明度的计算，并对初步 Mask 值进行提纯；最后通过平滑处理去除初步抠像结果中的噪声干扰，即可获得最优的数字抠像结果。

根据上述内容，结合 Kinect 中彩色图与深度图之间的一一映射关系，本文中的 Shared Matting 算法流程如下，其分为以下三个模块：

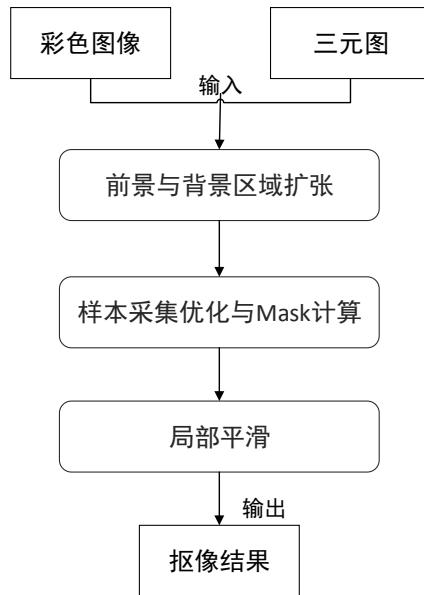


图 3-24 Shared Matting 算法流程

3.4.1 基于深度改进的区域扩张

如算法流程中所示，其将 3.3 节中获取的自动生成的三元图作为输入，区域扩张的思想为利用相邻像素之间的相似关系，确定像素的相似度从而减少未知像素的个数，即将三元图中“误分”到未知区域的前景或者背景点扩张为已知点。其具体操作如下：逐一检测未知区域内的像素点，根据条件判断当前未知像素是否为已知像素，若是，则将其扩张到对应到前景或是背景区域，否则保留。

原始算法中两个像素之间的复杂度以对应彩色图中两个像素点的“空间距离”进行计算。对于两个像素 p 和 q ， $D_{image}(p,q)$ 和 $D_{color}(p,q)$ 分别表示两个像素之间的图像空间距离和颜色空间距离。对于未知区域内的任意像素 p ，如果存在一个已知区域内的点 q 满足 $D_{image}(p,q) \leq k_i$ ， $D_{color}(p,q) \leq k_c$ 且与像素 p 对应的 $D_{image}(p,q)$ 最小，则将点 p 标记为“误分”的点，即其是属于已知区域的点却被错误地划分在未知区域中。经验参数 k_i 取决于未知区域的大小，这意味着分辨率大的图像很可能需要大的 k_i 。实验统计证明，当 $k_i = 10$ 像素， $k_c = 5/256$ （用 RGB 空间的欧氏距离测量）时，对于大多自然图像具有较理想的抠像结果。

而结合本文中结合 Kinect 深度信息的抠像目标，提出了基于深度改进的区

域扩张算法。考虑到在彩色图像中,当前景与背景交界处的像素色值很接近时(如图 3-22 中红色线框内的部分),其无法被上述两个条件阈值判断为扩张像素点,原始的区域扩张条件就失去了作用,因此需要加入深度图信息作为第三个阈值判断条件,即未知像素与已知像素之间的深度距离 $D_{d-color}(p, q) \leq k_d$ 。在本文中,将彩色图的图像空间距离、颜色空间距离和深度图像的深度信息结合,从而对于前景和背景边缘颜色相近的区域也可以进行前景和背景的区分。

在经过基于深度改进的未知区域扩张后,减少了部分“误分”的已知像素,从而减少后续的采样和掩膜计算量。

3.4.2 样本采集优化与 Mask 计算

Shared Matting 算法的关键思想在于相似像素点之间共享样本点对来减少计算量。Shared Matting 延用了 Robust Matting 的“信任系数”这一概念来选择样本点对,同时加入更多考虑因素作为判断依据。即在样本点对的选择中,Shared Matting 更注重样本的精度而非一味追求数量的减少。

样本采集与 Mask 计算过程包含样本采集与提纯, Mask 计算与优化两个子过程。

1) 样本采集与提纯

样本点采集中,对于未知像素 $p \in T_u$,以 p 为起点发射出 k_g 条射线,将图像划分为等角度不重合的扇区,起始射线被赋予一个相对于水平线的初始角度 $\theta \in [0, \pi/2]$,在 3×3 的窗口内每一个像素 p 有不同的初始角度,如图 3-25 所示。每条经过 p 的直线,在三元图的已知区域中,空间距离最小时至多取到一个前景像素和一个背景像素,因此该 k_g 条直线最多可得到 k_g 个前景像素和 k_g 个背景像素,会形成 k_g^2 个前背景样本点对。

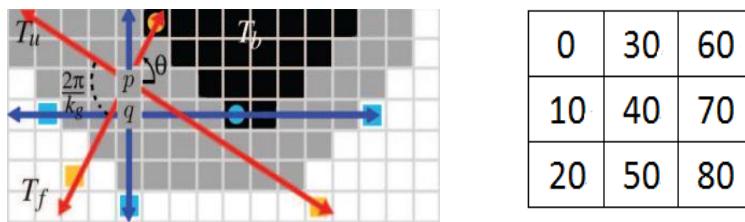


图 3-25 Shared Matting 中前背景样本的获取 (左) 及对应窗口中 θ 角度 (右)

得到 k_g^2 个样本点对之后,Shared Matting 引入了结合颜色信息,空间信息和概率密度信息的目标函数 g_p ,以此选取最优的前景背景样本点对,这个函数综合考虑了。如有一对前景背景样本点对 f_p 和 b_p ,前景颜色为 F_i ,背景颜色为 B_j ,则该目标函数:

$$g_p(f_i, b_j) = N_p(f_i, b_j)^{eN} A_p(f_i, b_j)^{eA} D_p(f_i)^{ef} D_p(b_j)^{eb} \quad (3-7)$$

其中, $N_p(f_i, b_j)$ 为像素 p 周围 3×3 邻域像素点的颜色畸变, 其定义了样本点对的可靠程度。由于相邻颜色空间上的像素具有线性聚类性质, 且在前后景的梯度模值小于透明度的梯度模值时, 两者的梯度成比例, 因此虽然未知像素 p 的透明度 α 取决于其本身尽可能小的颜色畸变 M , 但好的样本点对同时要求 p 周围邻域内的像素颜色畸变值也达到最小。

$A_p(f_i, b_j)$ 表示像素 p 计算得到的透明度估计值 α_p , 即 p 属于前景的概率相关。

$D_p(f_i)$ 和 $D_p(b_j)$ 为空间距离约束, 其表明 p 点距离其样本点对之间点距离越近约好。

常量 $e\{N, A, f, b\}$ 上式中每个因素的权重。经过实验经验表明, 当 $eN = 3$, $eA = 2$, $ef = 1$, $eb = 4$ 时, 目标函数可得到较好的计算结果。

因此, 通过估计目标函数 $g_p(f_i, b_j)$, 可得到像素点 p 点最佳前后景匹配样本对:

$$(\hat{f}_p, \hat{b}_p) = \arg \min_{f, b} g_p(f_i, b_j) \quad (3-8)$$

假设最优前后景样本点对 (\hat{f}_p, \hat{b}_p) 的颜色值为 (F_p^g, B_p^g) , 则计算样本点对在各自像素邻域 Ω_f 和 Ω_b 的颜色方差 σ_f^2 和 σ_b^2 :

$$\sigma_f^2 = \frac{1}{N} \sum_{q \in \Omega_f} \|C_q - F_p^g\|^2 \quad (3-9)$$

$$\sigma_b^2 = \frac{1}{N} \sum_{q \in \Omega_b} \|C_q - B_p^g\|^2 \quad (3-10)$$

其中 Ω_f 和 Ω_b 分别为以前景样本点 \hat{f}_p 和背景样本点 \hat{b}_p 为中心的 3×3 邻域, 其中 $N = 25$ 。

因此对于像素 p , 在样本采集阶段的输出为 $\tau_p^e = \{F_p^g, B_p^g, \sigma_f^2, \sigma_b^2\}$ 。

2) 样本点 Mask 的计算与优化

样本采集后, 针对 k_g^2 较小的像素点, 需要通过共享最优样本点对进行优化处理。

由于相邻未知像素之间在前景、背景以及透明度上的相似性, 在优化阶段, 对于未知区域中的像素 p 和相邻的 k 个像素点, 将其中颜色畸变 M 最小的三个最优样本点对的 RGB 三个分量求平均, 从而得到关于像素 p 的新变量 $\hat{\tau}_p^r = \{\hat{F}_p^r, \hat{B}_p^r, \hat{\sigma}_f^2, \hat{\sigma}_b^2\}$, 均值化的意义在于降低计算 Mask 的过程中透明噪声的产生。

经过优化后的像素 p 具有新的 $\tau_p^r = \{F_p^r, B_p^r, \alpha_p^r, f_p^r\}$:

$$F_p^r = \begin{cases} C_p & \text{if } \|C_p - \tilde{F}_p^g\| \leq \tilde{\sigma}_f^2 \\ \tilde{F}_p^g & \text{otherwise} \end{cases} \quad (3-11)$$

$$B_p^r = \begin{cases} C_p & \text{if } \|C_p - \tilde{B}_p^g\| \leq \tilde{\sigma}_b^2 \\ \tilde{B}_p^g & \text{otherwise} \end{cases} \quad (3-12)$$

$$\alpha_p^r = \frac{(C_p - B_p^r) \cdot (F_p^r - B_p^r)}{\|F_p^r - B_p^r\|^2} \quad (3-13)$$

$$f_p^r = \begin{cases} \exp\{-\lambda M_p(\tilde{F}_p^r, \tilde{B}_p^g)\} & \text{if } F_p^r \neq B_p^B \\ \varepsilon & \text{if } F_p^r = B_p^B \end{cases} \quad (3-14)$$

由上面四个式子可知：对于像素 p ，若其颜色值 C_p 与平均后的前景颜色值 \tilde{F}_p^r 相近，则设置其前景颜色值 F_p^r 为 \tilde{F}_p^r ，背景值 B_p^r 同理。 α_p^r 为 p 在上述最优样本点对下计算所得的透明度。而 f_p^r 为像素 p 在前景 F_p^r 和背景 B_p^r 上的置信度，前景和背景值不满足像素颜色 C_p 的模型时， f_p^r 的值很小。根据[13]中的实验结果，当 $\lambda=10$ 时实验测试结果较为理想。

对于已知区域中的前景像素，令其 $\tau_p^r = \{F_p^r, B_p^r, \alpha_p^r, f_p^r\} = \{C_p, C_p, 1, 1\}$ ，而对于已知的背景 $\tau_p^r = \{F_p^r, B_p^r, \alpha_p^r, f_p^r\} = \{C_p, C_p, 0, 1\}$ 。

3.4.3 局部平滑

样本点的采集和优化生成的 Mask 图虽然考虑了相邻未知像素之间的关系，但图像中仍存在透明度之间不连续的问题。因此，局部平滑既改善了透明度的局部平滑性，又保持了区域特性。局部平滑即对像素 p 以及其相邻的 n 个像素点的计算结果向量进行加权平均。

由上述步骤得到，像素 p 的最终前景 F_p 和背景 B_p 如下：

$$F_p = \frac{\sum_{q \in \psi_p} [W_c(p, q) \alpha_q^r F_q^r]}{\sum_{q \in \psi_p} [W_c(p, q) \alpha_q^r]} \quad (3-15)$$

$$B_p = \frac{\sum_{q \in \psi_p} [W_c(p, q) (1 - \alpha_q^r) B_q^r]}{\sum_{q \in \psi_p} [W_c(p, q) (1 - \alpha_q^r)]} \quad (3-16)$$

其中，

$$W_c(p, q) = \begin{cases} G(D_{image}(p, q)) f_q^r |\alpha_p^r - \alpha_q^r| & \text{if } p \neq q \\ G(D_{image}(p, q)) f_q^r & \text{if } p = q \end{cases} \quad (3-17)$$

上式中， G 为标准的高斯核函数。权值 W_c 综合考虑了像素点之间的空间关

系，像素前后景样本对之间的置信度以及透明度的差异。从而可以在新的样本点对前景色 F_p 和背景色 B_p 下求得最终对置信度 f_p 。

$$f_p = \min \left(1, \frac{\|F_p - B_p\|}{D_{FB}(\psi_p)} \right) \exp \left\{ -\lambda M_p(F_p, B_p) \right\} \quad (3-18)$$

其中：

$$D_{FB}(\psi_p) = \frac{\sum_{q \in \psi_p} [W_{FB}(q) \|F_q^r - B_q^r\|]}{\sum_{q \in \psi_p} W_{FB}(q)} \quad (3-19)$$

$$W_{FB}(q) = f_q^r \alpha_q^r (1 - \alpha_q^r) \quad (3-20)$$

得到 F_p , B_p 以及 f_p 之后，即可计算像素 p 的最终透明度 α_p 。

$$\alpha_p = \frac{(C_p - B_p) \cdot (F_p - B_p)}{\|F_p - B_p\|^2} + (1 - f_p) \alpha_p^l \quad (3-21)$$

其中， α_p^l 为低频透明度，其表示邻域中透明度的加权平均，其由下式(3-22)而得：

$$W_\alpha(p, q) = f_q^r G(D_{image}(p, q)) + \delta(q \notin T_u) \quad (3-22)$$

$$\alpha_p^l = \frac{\sum_{q \in \psi_p} [W_\alpha(p, q) \alpha_q^r]}{\sum_{q \in \psi_p} W_\alpha(p, q)} \quad (3-23)$$

$$\delta \text{ 为二值函数，其取值： } \delta = \begin{cases} 1 & \text{if } q \notin T_u \\ 0 & \text{else} \end{cases}$$

综合上文，经过 Shared Matting 计算之后的像素 p 有其对应的一组向量 (F_p, B_p, α_p) 以及其最终置信度 f_p 。

下图为不同图片在自动生成三元图的 Shared matting 中的测试结果：
实验 1：





(d)抠像得到的 Mask 图

(e)抠像所得的前景

(f)新的“人景合一”的图

实验 2:



(a)原图

(b)二值化图

(c)自动生成的三元图



(d)抠像得到的 Mask 图

(e)抠像所得的前景

(f)新的“人景合一”的图

图 3-26 自动生成三元图的 shard matting 抠像结果

本节主要介绍 Shared Matting 的算法过程，经过基于深度改进的区域扩张，样本采集和 Mask 计算优化以及区域平滑三个步骤，最终获得了原图像中每个像素点对应的最优样本点对以及透明度 α ，得到一幅与原图像大小相等的 Mask 图。在该算法中，由于各计算步骤相互独立，可以并行运算，从而在共享样本点减少大量运算量的基础上进一步提高了抠像效率，以达到实时抠像的目的。

3.5 基于 Kinect 的抠像算法测试与分析

3.5.1 算法流程

结合本章 3.2 中所表述的基于深度图的引导滤波算法，3.3 中提出的三元图

自动生成方法以及 3.4 中阐述的基于改进的 Shared Matting 抠像算法，本文提出的基于 Kinect 的抠像算法，其主要算法流程如下图：

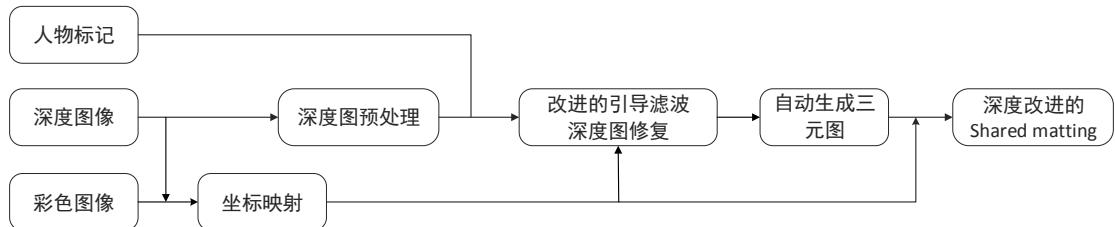


图 3-27 基于 Kinect 的抠像算法流程图

在完整算法中，首先通过 Kinect 获取实验所需的彩色图与深度图，接下来进行如下处理：

- 坐标映射：**由于同一时刻对应的彩色图和深度图分辨率不一致，首先对其进行坐标映射，使其位于同样的坐标系下并拥有相同的分辨率，以进入下一步中的引导滤波进行处理。
- 基于深度信息的引导滤波：**将经过坐标映射之后的彩色图作为引导图像，对深度图进行引导滤波，使其保留彩色图像对边缘信息，并初步修复 Kinect 深度图中存在的噪声、“空洞”和边缘残缺等问题。此步骤可得到基本完整且边缘较为平滑的深度图像。
- 自动生成三元图：**减少算法中的人工交互，并且为实时的视频抠像作铺垫，实现了基于形态学运算的自动生成三元图方法，针对本文抠像算法研究中前景的特性，利用形态学运算快速地生成一幅与深度图具有相同分辨率的三元图（Trimap）。
- 基于深度图的 Shared Matting 算法：**将经过步骤 c 之后获得的三元图和经过 a 坐标映射之后的彩色图作为算法的输入，经过改进的区域扩张步骤之后，采用共享样本点方法对彩色图和三元图进行抠像计算，最终获取较为精确的抠像结果。

3.5.2 算法平台与测试结果

本实验在 PC 平台上完成，结合 Kinect 二代传感器的特性，PC 的配置信息为：

处理器：Intel(R) Core(TM) i5-3470 CPU @ 3.2GHz

安装内存（RAM）：4.00GB

操作系统：Windows 8.1 专业版

算法在 Matlab R2013b 中进行仿真，最终在 VS2013 平台上完成，编码过程中结合了 Kinect SDK 与 OpenCV3.0，采用一系列自然图像进行测试后，得到如下算法结果。

针对本文所研究基于 Kinect 的人物抠像系统，主要对真实环境中带有人物前景的图像进行了测试。本文中的自动生成三元图算法人为地扩大了未知区域，而由于本文中每一帧彩色图像都有对其对应的深度图像，因此文中基于深度改进的区域扩张在一定程度上减少了未知点的像素数目以提升后续的抠像效率。本文中在区域扩张中加入的基于深度信息的扩张条件 $D_{d-color}(p, q) \leq k_d$ ，将 k_d 设定为 5/256。

基于深度改进的区域扩张与原始的区域扩张测试结果对比如下：

表 4-1 RGB 与 RGB-D 区域扩张处理结果对比

	原始的 RGB 区域扩张		本文 RGBD 改进的区域扩张	
	扩张点数	处理时间(s)	扩张点数	处理时间(s)
实验 1	8920	0.0352	10200	0.0396
实验 2	12504	0.0420	14932	0.0485
实验 3	9532	0.0370	12410	0.0434

由上表可知，基于深度改进的区域扩张能找到更多被“误分”为未知区域的像素点，且没有增加过多的时间复杂度。基于深度改进的 Shared Matting 在前景与背景有叠加的边缘区域具有更好的抠像精度。为验证本文研究的基于 Kinect 的抠像算法的抠像效果，本文利用 Kinect 采集了室内场景中的多组图像，其测试结果如下：

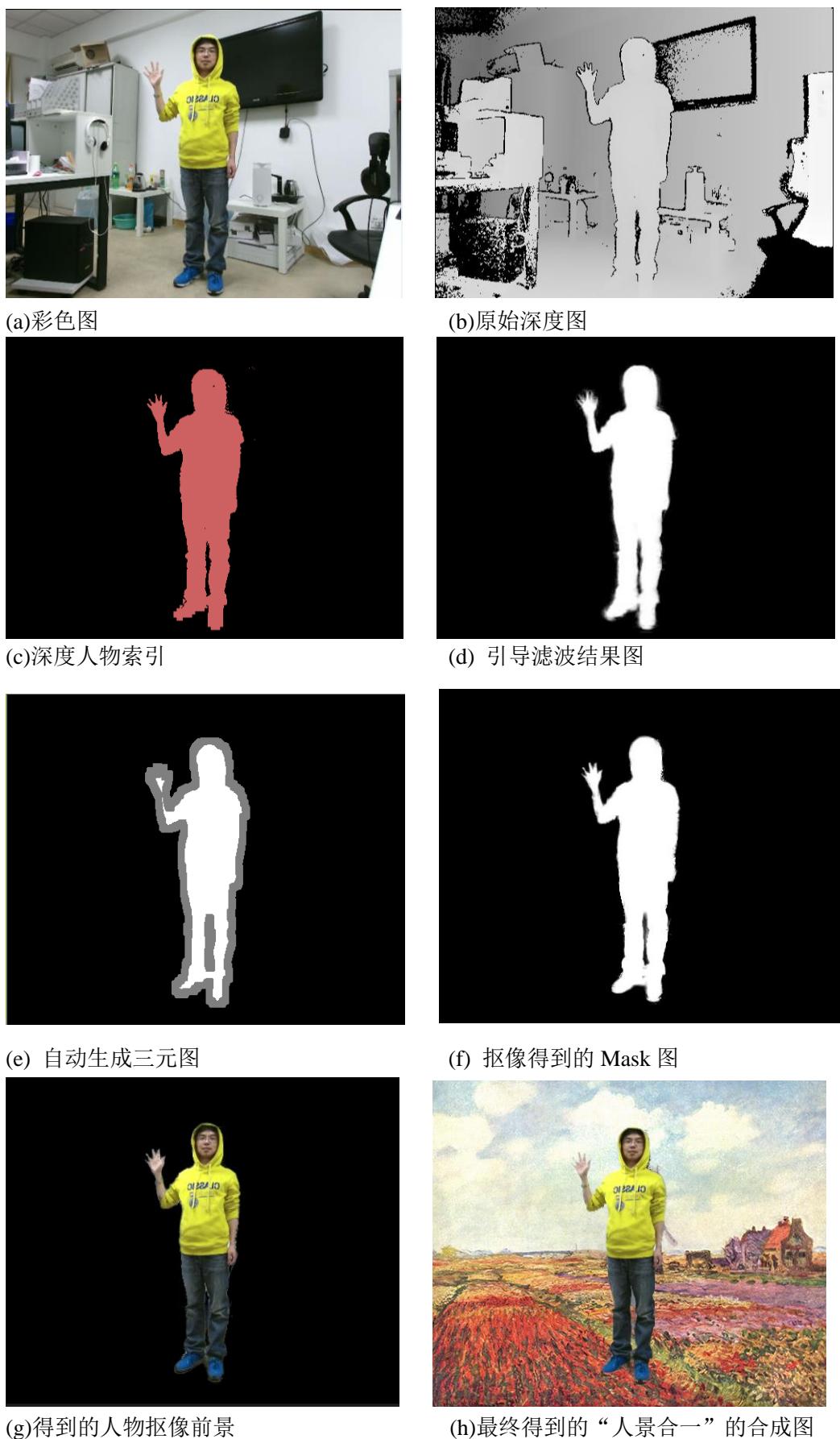


图 3-28 实验一：人物全身进行抠像结果

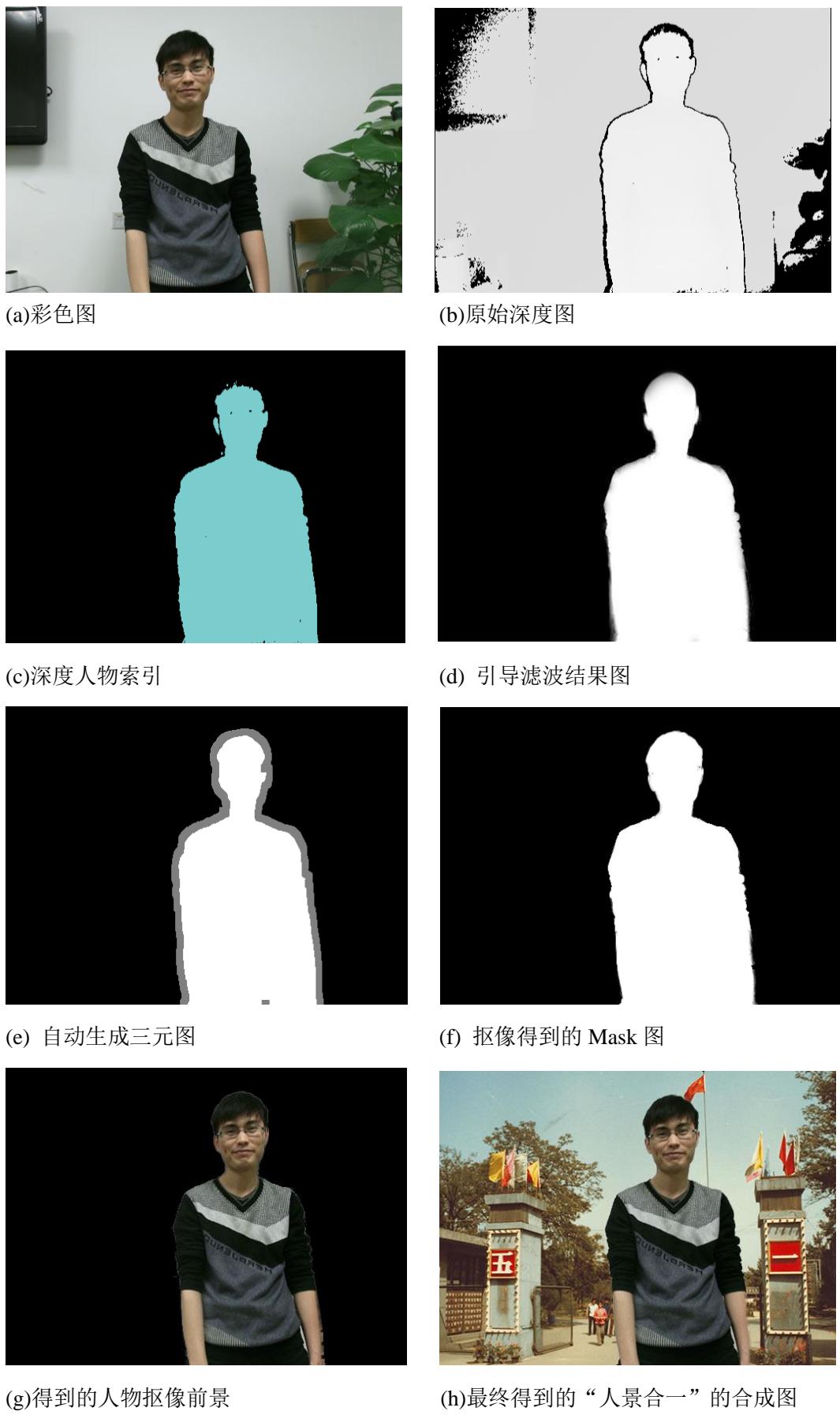


图 3-29 实验二：不带手势的人体半身抠像结果

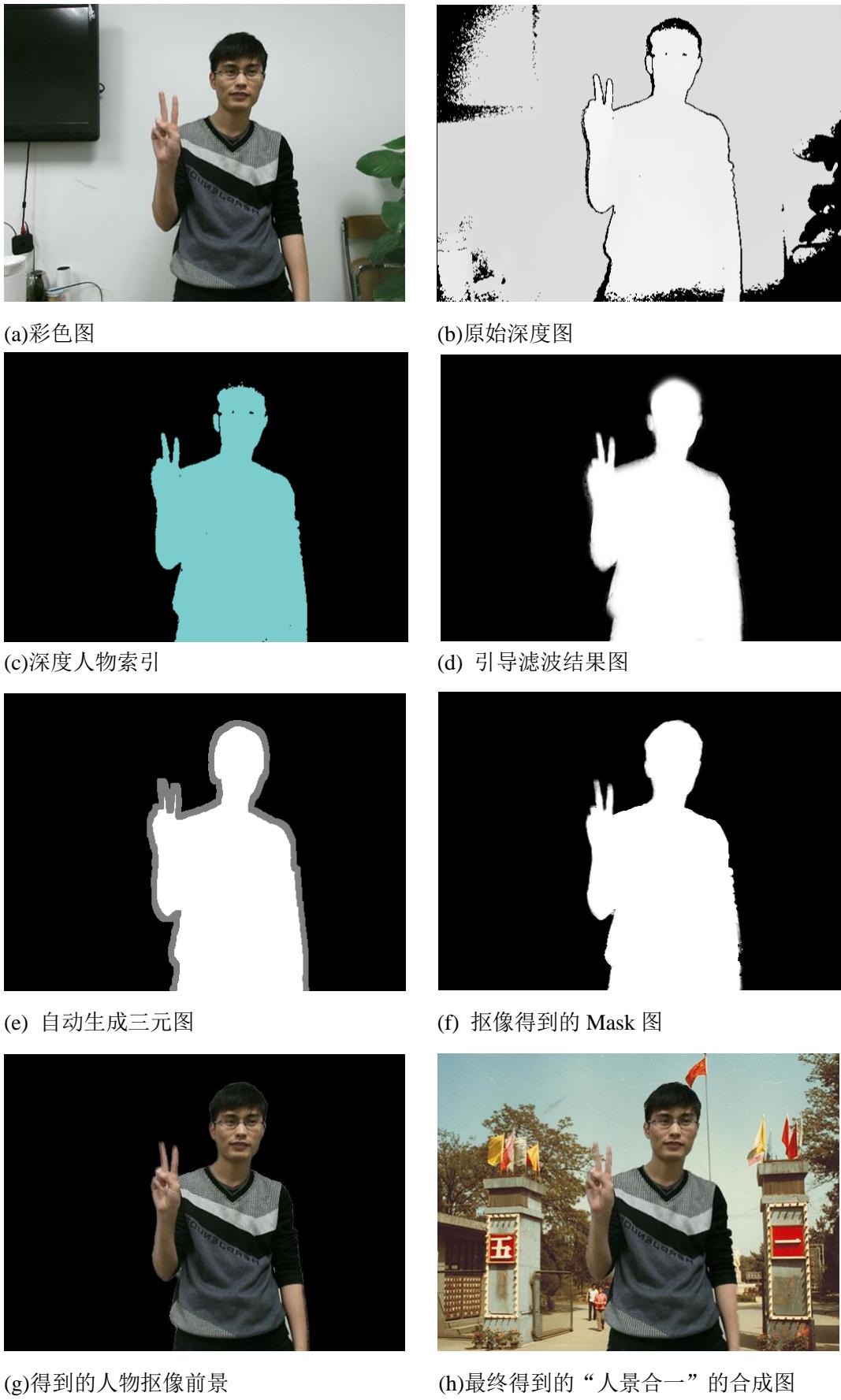


图 3-30 实验三：带手势的人体半身抠像结果

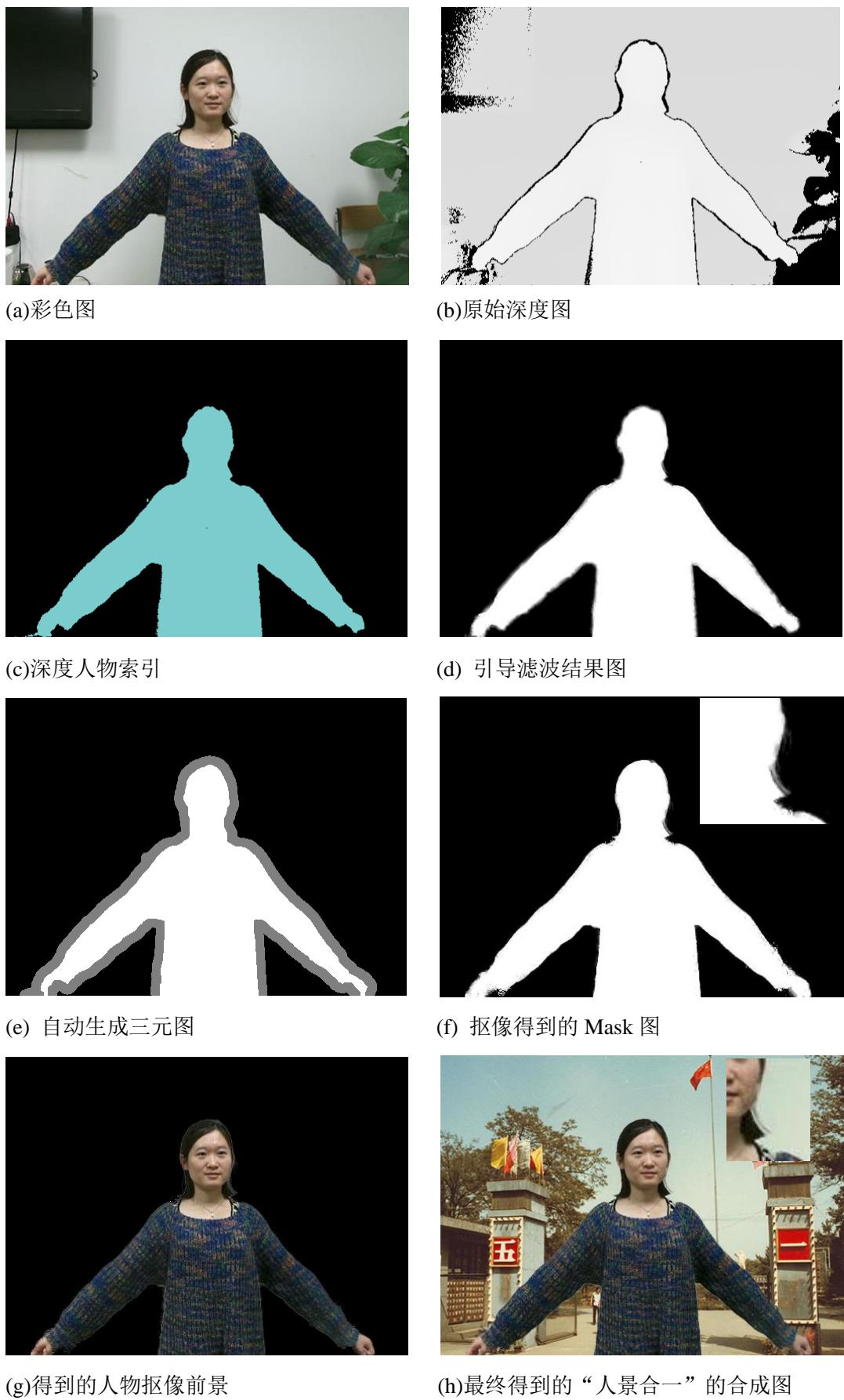


图 3-31 实验四：带发丝细节的人体半身抠图结果



图 3-32 实验五：带有前景附加小物件的人像抠取结果

如几组测试图所示, Kinect 获取场景中的彩色图以及对应的深度图像后, 首先经过基于深度图的引导滤波器对 Kinect 深度图中的“空洞”以及边缘的残缺进行基本的修复, 同时保证其与彩色图有相似的边缘特性, 然后通过自动生成三元图算法生成带有未知区域的三元图作为基于深度信息优化的 Shared matting 的输入图像, 结合 Kinect 彩色图进行最后的数字抠像处理。经过测试, 本文中基于 Kinect 的抠像算法对于不同的人物前景都有较好的抠像结果, 在人物的帽子, 右侧肩膀的衣服以及鞋子边缘都变得更为精细和平滑, 同时对于头发等较细节的地方也能得到较完整的抠像结果。最终抠取得到的人物前景能够与新的背景达到较为理想的“人景合一”的效果。

3.6 本章小结

本章作为本文的研究重点, 系统地阐述了 Kinect 的工作原理, 对两代 Kinect 进行了硬件和性能上的对比和总结, 同时分析了 Kinect 在进行深度信息测量时产生的误差成因及其解决方法, 阐述了 Kinect 中彩色图像与深度图像之间的坐标映射原理及方案。

针对 Kinect 深度图中存在的噪声, “空洞”, 边缘锯齿等问题, 本章提出了基于 Kinect 深度图的引导滤波器, 并将其迭代使用以达到较好的实验效果。

在利用引导滤波器对 Kinect 深度图进行预处理之后, 本章提出了一种自动生成三元图的方法, 省去了传统数字抠像过程中繁琐的人工交互步骤, 并为实时的视频抠像算法提供支持。最后本章介绍了一种支持实时抠像的 Shared Matting 算法, 并对其区域扩张步骤进行改进, 提出了结合了深度图空间距离权重的计算方法。最终利用改进的 Shard matting 完成基于 Kinect 的抠像算法, 并获得较好的效果。为后文中要搭建的实时抠像系统提供理论基础。

第四章 基于 Kinect 的抠像系统的设计与实现

本文在第三章所提出的基于 Kinect 的抠像算法的理论基础上，设计并实现了一个可视化的基于 Kinect 的实时抠像系统。系统能实时获取 Kinect 的彩色图与对应的深度图，并结合抠像算法展示实时的抠像效果并与新的背景图进行合成，为得到更自然的图像融合效果，在抠像功能的基础上，本系统还添加了实时的色彩匹配功能，并可供用户进行实时的视频流人物抠像与合成存储，得到新的“人景合一”的图片。本章将介绍整个系统的框架设计，开发环境以及每个功能模块的具体实现过程，最终给出系统的运行结果并进行分析。

4.1 系统组成与工作流程

4.1.1 系统介绍

基于 Kinect 的抠像系统主要由三大功能模块组成：实时抠像与合成、抠像结果与新背景之间的颜色匹配以及最终图片的存储。本系统利用 Kinect 实时捕获场景中的 RGBD 视频流，首先运用基于 Kinect 的抠像算法结合深度信息对场景中的彩色数据进行抠像处理，并与外部输入的虚拟背景相融合，接着加入色彩匹配功能，将新的背景图的色调迁移至人物抠像结果中，使去除了原始背景的人物抠像结果与新的背景图更自然地融合。考虑到系统运行的效率和性能，本文采用 Direct2D 对系统获取对图像进行渲染，在得到新的人景合一的合成图像后，通过存储功能对当前图像帧进行保存。图 4-1 描述了本系统包含上述三个模块在内的工作流程。

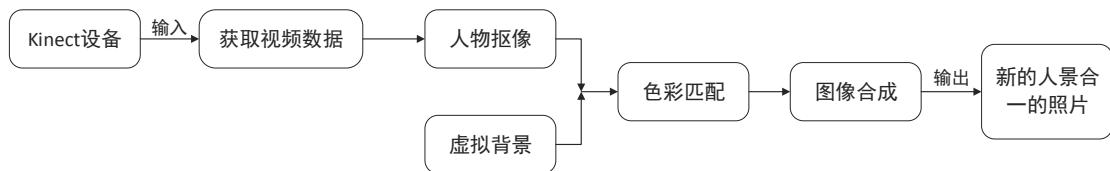


图 4-1 系统工作流程

4.1.2 系统架构

本系统的系统架构如图 4-2 所示：

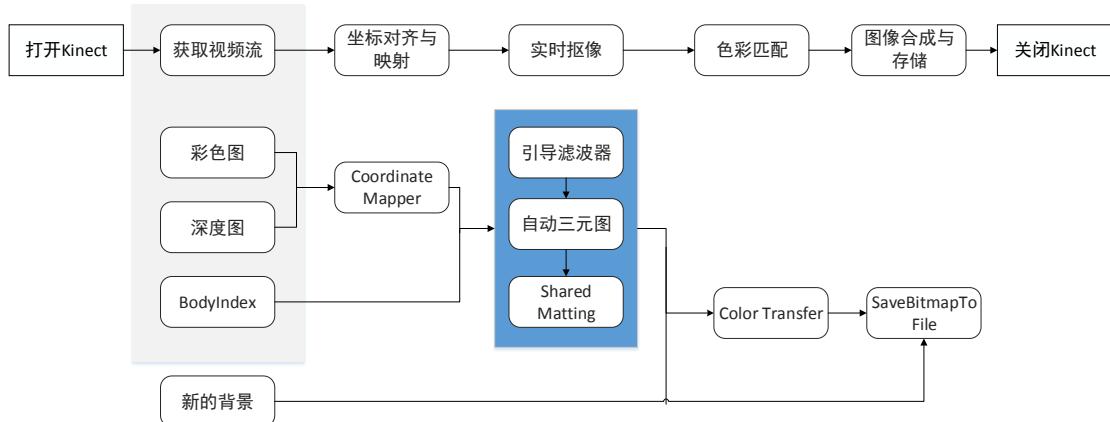


图 4-2 基于 Kinect 的抠像系统的系统架构图

根据系统架构，本文基于 Kinect 的抠像系统的实现主要分为四部分：

- ① Kinect 彩色和深度视频流的获取和预处理
- ② 基于 Kinect 的实时抠像算法
- ③ 色彩匹配功能算法
- ④ 图像合成与存储。

4.2 系统的开发环境

本文系统主要在 Visual Studio2013 上搭建，结合 Kinect 传感器以及其 SDK，结合 OpenCV 进行算法的研究和实现，结合 Direct2D 对算法图像进行渲染并保存输出，最终利用 MFC 生成可视化的系统操作界面。

4.2.1 Kinect for Windows SDK

Kinect for Windows SDK 是微软发布的与 Kinect for Windows 配套使用的软件开发工具包，其为 PC 提供 Kinect 的相关驱动，用以控制 Kinect 的工作流，具有强大的音视频流获取，骨骼识别跟踪以及手势识别等功能。其为基于 Kinect 的应用的研究和开发提供了大量的 API 和工具包，支持开发者使用 C#，VB，JS，C++ 等多种语言进行应用层上不同平台应用的开发，以帮助开发者们理解 Kinect 的工作机制，并将其用于应用的开发和完善。

在推出新生代 Kinect 之后，微软于 2014 年 8 月推出了 Kinect for Windows SDK V2 preview，在 Kinect for Windows SDK 1.x 系列的基础上更新了 200 多项改进，全面提升了彩色、深度、骨骼以及音频数据流，同时添加了更多的手势，还包括强大的 Kinect Fusion 工具包，以及更高分辨率的相机追踪，甚至允许多个应用同时使用传感器。于此同时其对于系统环境的硬件要求和软件环境都有了更高的要求。

本文将使用 Kinect for Windows SDK 2.0 来获取传感器中的彩色和深度视频流，并利用 SDK 中提供的坐标映射等函数对图像数据进行处理。

4.2.2 OpenCV

OpenCV^[47]全称 Open Source Computer Vision Library。其由 Intel 公司于 1999 年建立，是一个跨平台的开源计算机视觉库，可以在多个操作系统上运行，如 Linux、Windows 和 Mac OS，它由一系列 C 函数和部分 C++ 类构成，提供了 Matlab、Java 和 Python 等多个语言的接口，提供了大量数字图像处理和计算机视觉领域的通用算法。其应用领域包括人机交互、物体识别、图像分割、人脸识别、运动跟踪和机器视觉等等，而 2014 年 11 月，随着 OpenCV 3.0 beta 的发布，其在保留 OpenCV2 经典编程风格的基础上大大增强和改善了对 Python 和 Java 接口的支持，同时进行了架构调整，增加了 TLD 和鱼眼镜头模型等全新算法，还包括汽车检测等高级封装，并且对更多指令集进了优化。OpenCV 3.0 的提出无疑为本文中的算法实现提供了极大的平台帮助。

4.3 Kinect 彩色图像与深度图像的获取

Kinect 的工作机制中，每个类型的数据有对应的三个类，分别为 Source，Reader 以及 Frame。其获取数据的工作流如图 4-3 所示：



图 4-3 Kinect 传感器获取数据的工作流

1) Source: 打开一个 Kinect 设备之后，即可向其请求打开一个源 (Source)，根据请求的 Source 的种类，Kinect 获取相应的帧源：如彩色图像 (ColorFrameSource)、深度图像 (DepthFrameSource)、红外图像 (InfraredFrameSource)、人物索引 (BodyIndexFrameSource) 或是骨骼数据 (BodyFrameSource)。

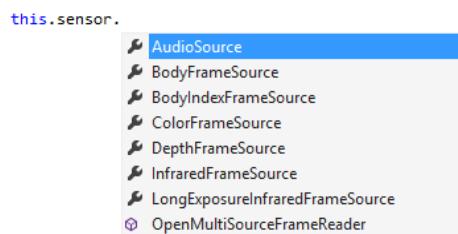


图 4-4 Kinect 中包含的 Source 种类

2) Reader: 由于 Source 为 Kinect 端所有，因此要获取创建一个读口 (Reader)，

与该 Source 绑定，而后通过调 Reader 来读取数据。每个 Source 可创建多个 Reader。

3) Frame: Frame 是真正存储数据的类。Reader 将数据读到 Frame 中，接着程序从 Frame 中提取需要的数据进行后续处理。

Kinect 总端口的类 IKinectSensor 提供了两个打开和关闭传感器的函数：

打开传感器：IKinectSensor::Open()

关闭传感器：IKinectSensor::Close()

Kinect 中获取彩色图的实现步骤：

1) 打开传感器

2) 获取彩色帧源：

如 IColorFrameSource *pColorFrameSource ColorFrameSource:

IKinectSensor::get_ColorFrameSource(IColorFrameSource **)

3) 打开一个彩色帧读取器：

ColorFrameReader *pColorFrameReader IColorFrameSource::OpenReader
(IColorFrameReader **)

4) 获取彩色帧：

IColorFrame *pColorFrame IColorFrameReader::AcquireLatestFrame
(IColorFrame**)

上述模式成为轮询模式，即在获取每帧之前都刷新已确定 Kinect 有没有新的彩色帧到达。

5) 获取彩色帧描述及数据参数：

IColorFrame::get_FrameDescription，该描述可用于获取彩色帧的长宽分辨率以及源数据格式。虽然目前 Kinect 的彩色图只支持 1920x1080 一种分辨率，不过考虑到系统的迁移性及升级维护，仍然用上述方法来获取彩色帧的长度和宽度。

6) 彩色帧的格式及存储：

Kinect 为彩色图像提供了多种格式，而在本系统中需要 RGBA 格式，因此需要首先对获取的彩色帧进行检查：

IColorFrame::get_RawColorImageFormat

若其格式为 ColorImageFormat_Bgra，则可直接使用以下函数获取数据地址与长度的 BGRA 位图：IColorFrame::AccessRawUnderlyingBuffer(UINT*, BYTE**);

若其格式不为 BGRX，则需要自备缓冲区进行转换：

IColorFrame::CopyConvertedFrameDataToArray(UINT, BYTE*,
ColorImageFormat)

至此，已将 Kinect 捕获的彩色图像存储至 RGBQUAD 格式的数组中，可以进行下一步的操作。

同样的，Kinect 中深度信息和人物索引模式的获取步骤与彩色信息获取的步骤类似。深度图的获取中，相应的 Source 类为 IDepthFrameSource，Reader 类为 DepthFrameReader，Frame 类为 IDepthFrame，而由于 Kinect 用一个 16 位的无符号深度值，Kinect 用一个 16 位无符号整型表示深度图像中上一个深度元素，单位是毫米。如 3-1 中，Kinect 中的有效距离为 0.5 米到 4.5 米（500, 4500），因此需要对其进行可视化处理，本文采用的方法为根据 Kinect 获取的当前灰度信息的最近和最远距离：

```
IDepthFrameSource::get_DepthMinReliableDistance  
(&unsigned short minDepth);  
  
IDepthFrameSource::get_DepthMaxReliableDistance  
(&unsigned short maxDepth);
```

在可视化时，根据每个深度元素的深度值与最大和最小深度之间的距离，将其标记为[0, 255]的灰度，即生成了一幅用灰度信息描述场景深度值的分辨率为 512x424 的图像。

本文设计的系统中，主要针对人像进行前景抠取，因此其在获取彩色图和深度图的同时，还需要获取包含了人物信息的人物索引信息，其获取步骤与深度信息的获取步骤一致，同样的只需将对应的三个类替换为：IBodyIndexFrameSource，IBodyIndexFrameReader 以及 IBodyIndexFrame 即可。在 Kinect 人物索引信息中，可识别六个人体，其用一个字节来表示检测到的人物标记，其取值有以下几种情况：

- ① -1: no body, 未检测到人体
- ② 0-5: 分别为检测到的 5 个人体标记
- ③ 其他: 未被使用

因此，进行人物索引信息的可视化，只需将人物标记为[0, 5]之间像素点标记为白色（255），其他部分为 0，即得到了带有人物索引的前景二值图。

而在本系统中，要获取同一时刻的彩色图、灰度图以及人物索引，需要同时开启彩色数据流，深度数据流以及人物索引数据流，而多个工作流的同步较为复杂，因此无法获知不同流到达的先后顺序。在 Kinect SDK 中提供了一个 MultiSourceFrameReader（复源帧）对象，将多个数据流封装在一起。

复源帧中同时获取彩色、深度和人物索引三种图像流：

```
IKinectSensor::OpenMultiSourceFrameReader(  
FrameSourceTypes::FrameSourceTypes_Color|
```

```
FrameSourceTypes::FrameSourceTypes_Depth|
FrameSourceTypes::FrameSourceTypes_BodyIndex,
IMultiSourceFrameReader**)
```

在获取多个数据流的 Reader 之后，获取复源数据帧：

```
IMultiSourceFrameReader::AcquireLatestFrame(IMultiSourceFrame**)
```

此时即可获取具体数据源的具体描述信息：

```
IMultiSourceFrame::get_ColorFrameReference
IMultiSourceFrame::get_DepthFrameReference
IMultiSourceFrame::get_BodyIndexFrameReference
```

通过上述方法，实现了本系统中的第一步——彩色图像及深度信息的获取，为保证后续算法的进行，需要对彩色图像与深度图像进行坐标映射，在 Kinect 中提供了一个坐标映射对象，能够对 Kinect 获取的多种数据之间进行坐标映射，比如将深度坐标映射至彩色图像坐标，将彩色图像映射至深度图像坐标等。在 ICoordinateMapper 类中定义了多个映射方法，本系统中为保证彩色图与深度图的一致性，并保证抠像算法的性能，将彩色图像坐标映射到深度图像：

```
ICoordinateMapper::MapColorFrameToDepthSpace(UINT
depthDataPointCount,  UINT16 *depthFrameData,  UINT depthPointCount,
DepthSpacePoint *depthSpacePoints)
```

通过上述方法，系统将彩色坐标映射到深度空间，即可进行后续的滤波，抠像等步骤。

4.4 抠像算法的实现

4.4.1 引导滤波的 OpenCV 实现

本系统中，在 VS2013 中使用 C++并借助 OpenCV 实现引导滤波器。本系统实现引导滤波器的关键思想在于通过积分图来实现的盒式滤波（box filter），以实现时间复杂度与窗口大小无关的算法，而 OpenCV 中的 boxfilter 函数满足这个要求。

Boxfilter 可以使原本复杂度为 $O(MN)$ 的求和或求方差等运算降低到近似于 $O(1)$ 的复杂度，而其缺点是不支持多尺度。因此本节首先对 BoxFilter 进行一个简单的介绍：

Boxfilter 的实现原理如下：

- ① 建立一个宽高为原图分辨率的数组 A
- ② 对数组赋值，元素 $A[i]$ 的值赋为该点邻域内的像素和（或平方和）

- ③ 在求解原图每个矩形内像素和的时，只需直接访问数组 A 中对应的位置即可，因此其复杂度为 $O(1)$ 。

OpenCV 中的 Boxfilter: `boxFilter(Mat sourceImg, Mat MeanImg, CV_32F, win_size)`

其中 `sourceImg` 为待计算均值的滤波图像，`MeanImg` 为均值运算结果，`CV_32F` 即在实现中需要将 `Boxfilter` 的待滤波图像扩展为 32 位浮点数。如图 4-5 所示：

```
void makeDepth32f(Mat& source, Mat& output)
{
    if (source.depth() != CV_32F) > FLT_EPSILON)
        source.convertTo(output, CV_32F);
    else
        output = source;
}
```

图 4-5 将待滤波图像扩展为 32 位浮点数

根据 3.2 中所阐述的引导滤波公式：

$$a_k = \frac{\frac{1}{|w|} \sum_{i \in w_k} I_i p_i - \mu_k \bar{p}_k}{\sigma_k^2 + \varepsilon} \quad (3-4)$$

$$b_k = \bar{p}_k - a_k \mu_k \quad (3-5)$$

a_k 的分子中， $\frac{1}{|w|} \sum_{i \in w_k} I_i p_i$ 即窗口 w_k 中每个像素 Ip 的和再除以窗口中像素的总个数，这即是一个简单的盒式滤波。因此本文中，在 OpenCV 实现引导滤波的过程中，先将输入的引导图像 I 和待滤波图像 p 相乘，并对相乘后的图像进行 `box filtering`。

分子中的第二项 μ_k 和 \bar{p}_k 分别为 I 和 p 在窗口 w_k 中的均值，即分别对 I 和 p 进行 `box filtering`，再将两者的结果相乘。而两者相减即为 Ip 在窗口 w_k 中的协方差。

而 a_k 中的分母中， σ_k^2 为引导图像 I 在窗口 w_k 中的方差，根据方差与均值之间的关系： $DX = E(X^2) - (EX)^2$ 。

因此在计算 I 的方差时，只需先计算 $I * I$ 的均值，并与 I 的均值 μ_k 的平方相减，而 $I * I$ 的均值计算与 Ip 的均值计算同理。最后在计算所得的方差图像的基础上加上分量 ε ，就完成了分母的计算，即得到 a_k 的值。

得到 a_k 之后， b_k 的值只需通过 p 在窗口 w_k 中的均值减去 a_k 与 I 在窗口中的均值乘积即可。

第二个公式：

$$q_i = \frac{1}{|w|} \sum_{i \in w_k} a_k I_i + b_k = \bar{a}_i I_i + \bar{b}_i \quad (3-6)$$

输出值 q_i 分别与 a 和 b 在窗口 w_k 中的均值有关，因此需要对第一步结果中所得的 a 和 b 再进行盒式滤波得到两个新图： a_i 和 b_i ，接着根据公式(3-6)，得到 a_i 与引导图像 I 的乘积，并于 b_i 相加，最终得到输出图像 q 。

根据上述实现步骤，在 OpenCV 中只需几十行代码即可将实现 Kinect 中人物深度图的引导滤波。

4.4.2 三元图的自动生成

经过引导滤波器后，可得到噪声明显被抑制且边缘基本平滑的人物前景二值图，在此基础上，利用 OpenCV 强大的形态学运算库进行自动的三元图生成。

① 首先分别定义用于进行腐蚀和膨胀运算的结构元素：

```
IplConvKernel*erode_element=
cvCreateStructuringElementEx(k1, k1, 0, 0, CV_SHAPE_RECT)
IplConvKernel*dilate_element=
cvCreateStructuringElementEx(k2, k2, 0, 0, CV_SHAPE_RECT);
```

根据不同的图像特性，可分别定义腐蚀和膨胀的结构元素大小和形状为相同或者不同。

② 膨胀运算：

```
cvDilate(pImage, pDilate, dilate_element, n);
```

其中 $pImage$ 为引导滤波之后的图像结果，其为一个二值图， $pDilate$ 为膨胀运算之后的结果图， n 为迭代次数，一般默认取 1。

③ 腐蚀运算：

```
cvErode (pImage, pErode, dilate_element, n);
```

其中 $pErode$ 为腐蚀运算的结果图，其他参数与膨胀运算相同。

④ 三元图的生成：

将②与③中的结果相减，并将结果标记为灰色，即得到了三元图的未知区域，与前景相加，得到最终的结果图。

```
cvSub(pDilate, pErode, mGradient, NULL);
IplImage* mGrayGradient = FillImageGray(mGradient);
cvAdd(mGrayGradient, pErode, mGrayGradient, 0);
```

4.4.3 基于深度改进的 Shared Matting

在 Shared Matting 的实验过程中主要有四个主体函数：

`expandKnown()`: 区域扩张

`refineSample()`: 样本采集与优化

`localSmoothx()`: 局部平滑

`getMatte()`: 生成最终的抠像结果 MASK 图

将前三个步骤计算所得的每个像素的透明度 α 写入最终的结果图像 `matte` 中。

4.5 色彩匹配算法的实现

色彩匹配是指把一幅图像 A 的色彩信息如色调等迁移到另一图像 B 中，并输出新的图像 C ，使最终得到的图像 C 既具有 B 的形状信息，又保留了 A 的色彩信息^[48]。一般将图像 A 称为目标图像或颜色图像，将图像 B 称为源图像或形状图像。

在本文中，为保证抠像之后的前景图像与新的背景更自然地融合，在系统中添加了色彩匹配这一步骤，当新的背景为老照片或是带有特殊色调的图片时，其与前景图像的融合会更真实，从而达到“人景合一”的效果。

本系统的色彩匹配采用了 Erik Reinhard 等人提出的经典色彩迁移算法^[49]。Reinhard 等人针对人类的图像感知原理提出了具有正交积的 *lab* 颜色空间，其中， l 为亮度通道， a 表示彩色的黄—蓝通道， b 表示红—绿通道，其将三个通道之间的相关性降到了最小，因此更符合人类的视觉感知系统。在 *lab* 基础上提出了 ColorTransfer 经典算法，Reinhard 等人先将目标图像和源图像都转化到 *lab* 空间进行统计对齐，而后通过目标图像与源图像之间均值和方差到匹配运算，并将其转换到 *RGB* 空间，最终达到颜色匹配的目的。

其基本公式如下：

$$\begin{aligned} l^* &= l - \langle l \rangle \\ \alpha^* &= \alpha - \langle \alpha \rangle \\ \beta^* &= \beta - \langle \beta \rangle \end{aligned} \tag{4-1}$$

$$\begin{aligned} l' &= \frac{\sigma_t^l}{\sigma_s^l} l^* \\ \alpha' &= \frac{\sigma_t^\alpha}{\sigma_s^\alpha} \alpha^* \\ \beta' &= \frac{\sigma_t^\beta}{\sigma_s^\beta} \beta^* \end{aligned} \tag{4-2}$$

其算法实现流程如下：

- 1) initialize(): 对目标图像（新的背景图）和源图像（抠像结果）进行初始化，将其转化为 32 位精度，得到目标图像的 RGB 向量数组 targetImg_32F，以及源图像的 RGB 向量数组 srcImg_32F。
- 2) 调用 cvtColor(srcImg_32F, srcImg_lab, CV_BGR2lab)，将源图像从 *RGB* 空间转化到 *lab* 空间，用同样的方法将目标图像从 *RGB* 空间转化到 *lab* 空间。
- 3) 调用函数 computerMeans()，计算目标图像和源图像在 *lab* 空间三个通道中分别的均值和标准差。
- 4) solve(): 根据公式(4-1)，记 data_rate[k] 为目标图像与源图像标准差的比值，由目标图像中每个像素减去图片均值 Us 再乘上 data_rate[k] 最后加上源图像的均值，计算目标图片经过色彩匹配之后的 *lab* 值，存储于 Mat 数组 result_lab 中
- 5) 调用函数 cvtColor(targetImg_32F, targetImg_lab, CV_lab2BGR) 将目标图像从 LAB 空间重新转化回到 RGB 空间，并存储于 Mat 类型的数组 result 中。

下图展示了色彩匹配算法的处理结果



图 4-6 色彩匹配算法结果图



图 4-7 本文研究图像的色彩匹配合成结果

由上图可以看到，经过 Reinhard 的色彩匹配算法之后，可以很好的将目标图像的色调迁移到源图像上而不影响源图像整体的色彩协调性，并且其对目标对象的多种色调都可进行效果较好的色彩迁移。经过色彩匹配的前景图像能更自然地与虚拟背景进行融合。因此，此算法可以很好的进行色彩匹配，并可用于本文的系统中。

4.6 图像的合成与存储

图像的合成与存储按钮通过界面上的 button 控件完成，在点击界面中到保存图片按钮后，调用函数 `GetScreenshotFileName()` 获取当前帧合成图像的存储名称，调用函数 `SaveBitmapToFile()` 将图像存储至当前路径下。

在本系统中，先调用 `LoadResourceImage()` 函数将 MFC 图像展示控件在运行后加载 Images 下新的背景图片，当无法加载时，默认设置背景为绿屏，即 `const RGBQUAD c_green = {0, 255, 0}`。

在通过抠像算法与色彩匹配算法得到拥有新背景颜色信息的前景图像之后，将 MFC 图像展示控件属于前景的区域显示为色彩匹配后的前景像素值，从而将获取的人物前景与虚拟背景合成为一幅新的图像。

本文系统默认将图像命名为：KinectScreenshot-CoordinateMapper-hh-mm-

ss.bmp，其中 hh-mm-ss 为当前的系统时间。

GetScreenshotFileName(szScreenshotPath, _countof(szScreenshotPath)) 具体实现：

```
WCHAR szTimeString[MAX_PATH];
GetTimeFormatEx(NULL, 0, NULL, L"hh'-mm'-ss", szTimeString,
_countof(szTimeString));
StringCchPrintfW(lpszFilePath, nFilePathSize, L"%s\\KinectScreenshot-
CoordinateMapping-%s.bmp", pszKnownPath, szTimeString);
```

图像存储函数：

```
SaveBitmapToFile(reinterpret_cast<BYTE*>(m_pOutputRGBX), nColorWidth,
nColorHeight, sizeof(RGBQUAD) * 8, szScreenshotPath);
```

① 首先定义所保存的 bmp 图像格式的信息头文件以及分辨率信息：

```
bmpInfoHeader.biSize      = sizeof(BITMAPINFOHEADER);
bmpInfoHeader.biBitCount  = wBitsPerPixel;
bmpInfoHeader.biCompression = BI_RGB;
bmpInfoHeader.biWidth     = lWidth;
bmpInfoHeader.biHeight    = -lHeight;
bmpInfoHeader.biPlanes    = 1;
bmpInfoHeader.biSizeImage = dwByteCount;
```

- ② 在磁盘中创建包含上述信息的文件
- ③ 在文件中写入位图的文件头和信息头
- ④ 写入像素的 RGB 数据
- ⑤ 关闭文件

4.7 实验结果与分析

本文搭建的基于 Kinect 的抠像系统主要包括实时抠像与合成、色彩匹配、图片存储和实时帧率状态显示功能。因此在系统主界面上主要包括三个区域：①视频流抠像与合成结果显示区，②实时状态显示栏，③图片存储按钮。如下图所示：



图 4-8-1 系统主界面：默认绿屏



图 4-8-2 系统主界面：添加虚拟背景



图 4-9 人物抠像与合成视频流截取

针对实际场景中的人物前景与背景之间存在的色调差异，经过色彩匹配后，得到的视频流图像截取如下：



图 4-10 色彩匹配视频流

图片显示区域的状态栏会显示 kinect 连接状态，Direct2D 渲染失败警告，实

时的帧率，以及图片保存状态和路径等：



图 4-11-1 实时状态显示栏（FPS 为实时帧率，Time 为运行时间/s）

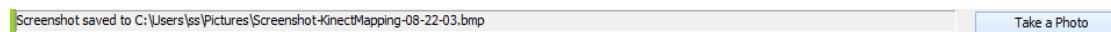


图 4-11-2 实时状态显示栏（图片存储状态+存储路径）

点击“Take a Photo”按钮后，经过合成的图片会自动保存进入默认路径：
C:\Users\username\Pictures\...

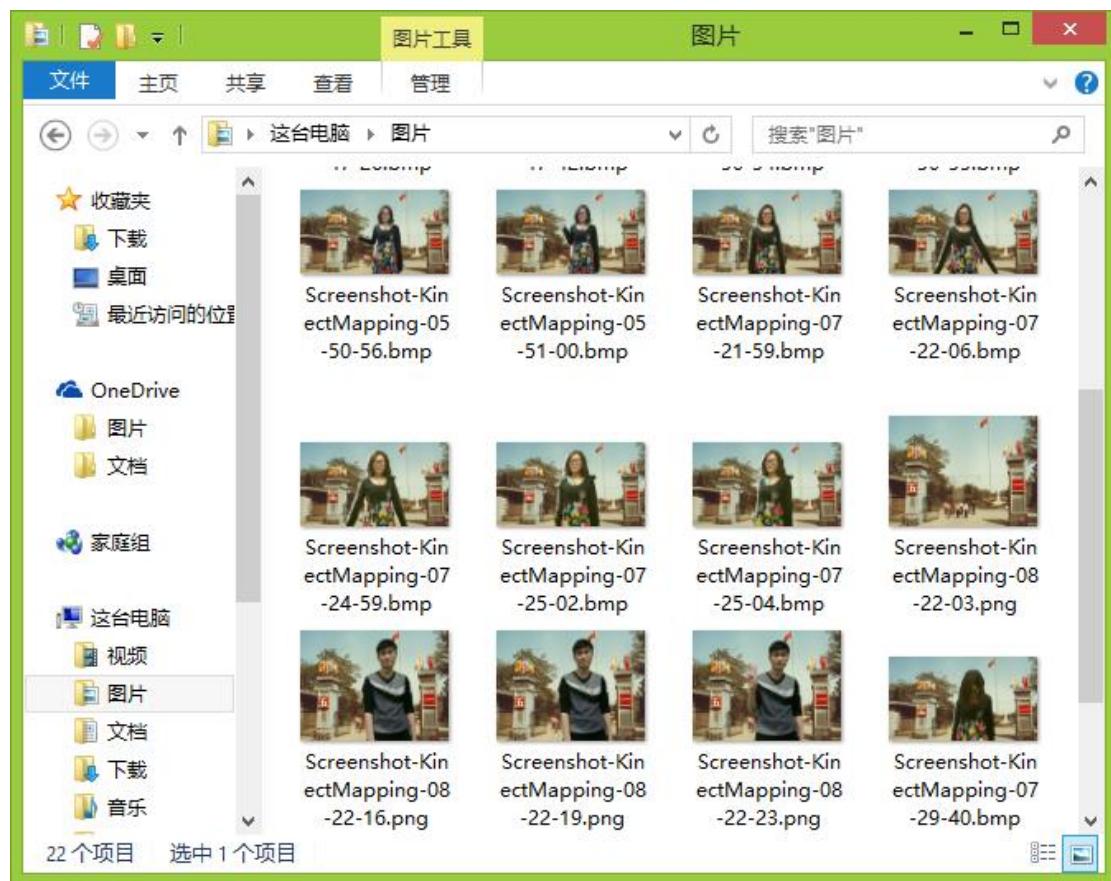


图 4-12 合成图像存储结果

4.8 本章小结

本章介绍了基于 Kinect 的抠像系统的设计与实现，首先根据系统架构介绍了每个模块的工作流程和模块任务，接着介绍了系统的开发工具 Kinect SDK 和 OpenCV，并且详细阐述了系统架构中每一步骤的实现过程，主函数和方法。最

终展示了本文搭建系统的实验结果，测试了每个功能模块并进行分析，证明本系统可很好地实现图像实时抠像与色彩匹配（15~30FPS），最终实现完整的图像合成和存储过程。

第五章 总结与展望

5.1 本文工作总结

数字抠像技术作为数字图像处理领域中越来越受到关注的课题，被广泛地应用于广告媒体行业、影视制作、游戏开发行业以及医疗卫生等等行业。本文的研究课题为基于 Kinect 的抠像算法的研究和应用，通过对 Kinect 工作原理以及基于深度图的数字抠像算法的学习和研究，本文主要完成以下几点工作：

- 1) 研究了体感外设 Kinect 的起源、发展和应用，比较并总结了一代和二代 Kinect 传感器硬件配置。并通过 Kinect 的硬件组成和数据流的介绍分析了 Kinect 的深度测量原理以及测量误差的主要来源，并研究了 Kinect 在彩色图和深度图在坐标映射上的不足以及解决方案。
- 2) 对比了几种传统经典抠像算法的原理以及抠像结果，并深入研究了基于 TOF 测距的深度抠像算法，将深度信息引入传统的 RGB 抠像算法中进行优化计算。针对本文研究的基于深度图的抠像算法，最终根据本文研究的实时性和抠像精度综合考虑选择了 Shared Matting 作为本文研究算法。
- 3) 研究了深度图像的平滑和滤波算法，将基于深度图改进优化的引导滤波引入到深度图像的处理中，并将其进行迭代使用以提高边缘和内部空洞处理的精度。
- 4) 利用 Kinect 获取的深度图像进行引导滤波处理后，实现了深度信息三元图的自动生成算法，作为抠像算法的输入图像之一，结合基于深度改进的 Shared Matting 算法实现了基于 Kinect 的实时抠像。
- 5) 实现了可视化的 Kinect 实时抠像系统，同时针对人物前景与虚拟背景色调不一致的问题，引入 Erik Reinhard 提出的经典的色彩迁移算法进行处理，最终能够得到前后景色调一致的新的“人景合一”的图像并进行保存。

5.2 不足与展望

本文目前实现的是初步的基于 Kinect 的抠像系统，由于时间关系和研究水平有限，本文在理论研究和实现上仍然存在一些不足，需要对其进行进一步的研究和探索。

- 1) 自然抠像中，原始图像会存在各种复杂的情况，本文研究的目标对象主要为室内场景，因此在研究过程中，没有考虑很多复杂情况，因此希望在之后的研究学习中可以改进算法对更复杂的图像进行准确的抠像处理。

2)本文采用了 Shared Matting 算法结合深度信息进行数字图像的抠像处理，但是如何设定更好的权重参数，更好地结合深度信息对传统的基于 RGB 的共享样本点进行改进，同时保证其实时性，需要使用 GPU 进行并行运算。需要对 Kinect 系统进行更好的优化处理，因此 Kinect 的研究仍然任重道远。

3) 本文搭建的 Kinect 抠像系统，实现简单的实时视频抠像中，为保证视频抠像的时间复杂度，进行了算法的简化处理，牺牲了一部分精度。并且对于 Kinect 抠像中，人物脚步与地面接触的区域，当颜色接近时，要取得很好的抠像效果，还需进一步研究。Kinect V2 目前作为体感外设，其深度图像分辨率 512×424 相比于高清彩色图有很大的差距，因此对于深度图像的上采样以及滤波处理上也有很大的提升空间。

4) 后续希望为 Kinect 抠像系统加入更多的交互元素，因为 Kinect 的开发本身支持丰富的体感交互。因此可以为本文系统添加手势切换背景，保存图片等，实现一个完全体感交互的抠像系统，其应用范围也会更广阔，同时有更多的趣味性和扩展性。

参考文献

- [1] 维基百科. Kinect_维基百科[G/OL]. <https://en.wikipedia.org/wiki/Kinect>, 2015.
- [2] 林生佑,潘瑞芳,杜辉等. 数字抠图技术综述[J]. 计算机辅助设计与图形学学报, 2007(04): 473-479.
- [3] 维基百科. 图像分割_维基百科[G/OL]. https://en.wikipedia.org/wiki/Image_segmentation, 2015.
- [4] 张展鹏,朱青松,谢耀钦. 数字抠像的最新研究进展[J]. 自动化学报, 2012 (10): 1571-1584.
- [5] Smith R, Blinn F. Blue screen matting[C]. In Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH, New Orleans, 1996: 259-268.
- [6] Chuang Y, Curless B, Salesin H, 等. A Bayesian approach to digital matting[C]. In Computer Vision and Pattern Recognition(CVPR), 2001 IEEE Conference on, 2001: 264-271.
- [7] Sun J, Li Y, Kang B. Flash matting [J]. ACM Transactions on Graphics, 2006, 25(3): 772-778.
- [8] Ruzon A, Tomasi C. Alpha estimation in natural images[C]. In Computer Vision and Pattern Recognition, 2000 IEEE Conference on, 2000: 18-25.
- [9] Chuang Y, Curless B, Salesin H, Szeliski R. A Bayesian approach to digital matting[C]. In Computer Vision and Pattern Recognition. 2001 IEEE Conference on, 2001: 264-271.
- [10] Juan O, Keriven R. Trimap segmentation for fast and userfriendly alpha matting [J]. Variational, Geometric, and Level Set Methods in Computer Vision, 2005, 3752: 186-197.
- [11] He M, Rhemann C, Rother C. A global sampling method for alpha matting[C]. In Computer Vision and Pattern Recognition. 11th IEEE Conference on, 2011: 2049-2056.
- [12] Sun J, Jia Y, Tang K. Poisson matting[C]. In Computer Graphics and Interactive Techniques (SIGGRAPH). 2004 ACM Conference on, 2004: 315-321.
- [13] Du L, Lin H, Qin Y 等. Oriented poisson matting[J]. In Image Processing, 2005. ICIP 2005. IEEE International Conference on, 2005: 626-629.
- [14] Bai X, Sapiro G. A geodesic framework for fast interactive image and video segmentation and matting[C]. In Computer Vision. 2011 IEEE International Conference on, 2007: 1-8.
- [15] Rhemann C, Rother C, Rav-Acha A 等. High resolution matting via interactive trimap segmentation[C]. In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, 2008: 1-8, 23-28.
- [16] Wang J, Cohen F. Optimized color sampling for robust matting[C]. In Computer Vision and Pattern Recognition. 2007 IEEE Conference on, 2007: 17-22.
- [17] Rhemann C, Rother C, Gelautz M. Improving color modeling for alpha matting [J]. British

- Machine Vision Conference. BMVA, 2008: 1155–1164.
- [18] Eduardo G, Oliveira M. Shared sampling for real-time alpha matting [J]. Computer Graphics Forum, 2010, 29(2): 575–584.
- [19] Zongker D, Werner D, Curless B. Environment matting and compositing[C]. In Computer Graphics Proceedings, 1999: 205-214.
- [20] Yeung K, Tang K, Brown S, 等. Matting and compositing of transparent and refractive objects[J]. ACM Transactions on Graphics, 2011, 30(1): 1–13.
- [21] Chuang Y, Zongker D, Hindorf J. Environment matting extensions: towards higher accuracy and real-time capture [C]. In Computer Graphics Proceedings, 2000: 121-130.
- [22] Matusik W, Pfister H, Ziegler R. Acquisition and rendering of transparent and refractive objects[C]. Proceedings of the 13th Eurographics workshop on Rendering, 2002: 267-278.
- [23] Chuang Y, Goldman D, Curless B. Shadow matting and compositing[C]. In Computer Graphics Proceedings, 2003: 494-500.
- [24] Apostoloff N, Fitzgibbon A W. Bayesian video matting using learnt image priors[C]. In Computer vision and Pattern Recognition, 2004 IEEE Conference on, 2004: 407-414.
- [25] Wang O, Jonathan, Yang Q. Automatic Natural Video Matting with Depth[J]. Computer Graphics and Applications, 2007: 469-472.
- [26] Jiejie Z, Miao L, Ruigang Y. Joint depth and alpha matte optimization via fusion of stereo and time-of-flight sensor[C]. In Computer Vision and Pattern Recognition(CVPR), 2009 IEEE Conference on, 2009: 453-460.
- [27] Wang J, Cohen, M.F. Optimized Color Sampling for Robust Matting [J]. Computer Vision and Pattern Recognition, 2007 IEEE Conference on, 2007: 1-8.
- [28] 何贝,王贵锦,林行刚. 结合 Kinect 深度图的快速视频抠图算法[J]. 清华大学学报(自然科学版), 2012(04): 561-565+570.
- [29] 夏倩,许勇. 基于 Kinect 的自动视频抠像算法[J].计算机工程与设计,2015(05):1299-1303.
- [30] 赵旭. Kinect 深度图像修复技术研究[D]. 大连: 大连理工大学, 2013.
- [31] 周星,高志军. 立体视觉技术的应用与发展[J]. 工程图学学报, 2010(04): 50-55.
- [32] Yang R, Cheng S, Yang W. Robust and Accurate Surface Measurement Using Structured Light[C]. Instrumentation and Measurement, 2008 IEEE Transactions: 1275-1280.
- [33] R.Gvili, A. Kaplan, E. Ofek. Depth Keying [C]. SPIE Electronic Imaging Conference, 2003.
- [34] Wang J, Cohen M. An iterative optimization approach for unified image segmentation and matting[C]. Proceedings of CCV, 2005: 936-943.
- [35] Guan Y, Chen W, Liang X. Easy matting[C]. Proceedings of Eurographics ,2006.
- [36] Yang Q, Yang R, Davis J. Spatial-Depth Super Resolution for Range Images. In Computer

- Vision and Pattern Recognition, 2007 IEEE Conference, 2007: 1-8, 17-22.
- [37] 张约伦. 基于 Kinect 的抠像算法研究[D]. 西安: 西安电子科技大学, 2013.
- [38] Cho J, Ziegler R, Gross M, 等. Improving alpha matte with depth information [J]. IEICE Electronics Express, 2009, 6(22): 1602-1607.
- [39] Han J, Ling S, Xu D, 等. Enhanced Computer Vision With Microsoft Kinect Sensor: A Review[J], IEEE Transactions 43(5): 1318-1334.
- [40] 丁津津. TOF 三维摄像机的误差分析及补偿方法研究[D]. 安徽: 合肥工业大学, 2011.
- [41] Rapp H. Experimental and Theoretical Investigation of Correlating TOF-Camera Systems [D]. Interdisciplinary Center for Scientific Computing (IWR): University of Heidelberg, 2007.
- [42] 陈理. Kinect 深度图像增强算法研究[D]. 湖南: 湖南大学, 2013.
- [43] Butkiewicz, T. Low-cost coastal mapping using Kinect v2 time-of-flight cameras [J]. Oceans St. John's, 2014: 1-9, 14-19.
- [44] Khoshelham K, Elberink S. Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications [J]. Sensors, 12(2): 1437-1454, 2012.
- [45] He K, Sun J, and Tang X. Guided image filtering [J]. Proc. 11th Eur. Conf. Comput. Vis. (ECCV), Hersonissos, Greece, 2010: 1-14.
- [46] Gonzalez R 等著, 阮秋琦等译. 数字信号处理(第二版)[M]. 北京: 电子工业出版社, 2003.
- [47] 维基百科 OpenCV_维基百科[G/OL]. <https://en.wikipedia.org/wiki/OpenCV>, 2015.
- [48] 郑宁斐, 姜光. 颜色迁移算法的研究[J]. 电视技术, 2006(11):89-90+94.
- [49] Reinhard E, Adhikhmin M, Gooch B. Color transfer between images[J], Computer Graphics and Applications, IEEE, 21(5), 2001: 34-41.

致 谢

在本课题的研究和论文撰写过程中，得到了很多人的帮助和指导，在此提出万分的感谢！首先要感谢我的导师李学明老师，在研究生就读的两年时间中，对我的诸多指导和帮助，特别是在本次毕业设计中的课题选取和研究方向上的悉心指导。李老师治学严谨，管理有方，对每一位学生都非常认真负责，关心我们的科研学习和生活各个方面。非常荣幸在我的本科和研究生学习生涯中都有李老师这样一位良师益友，希望在走出校门后，仍然可以带着这份教诲继续学习和提升自己。

同时要感谢实验室的同窗薛明、刘菲朵、倪尧天、温雅、周倩和麻家琪同学。在课题的开发过程中，薛明同学给了我很多帮助和指导，助我快速地找到问题的解决方案。两年半的共同学习和生活，我们并肩奋斗，本着踏实进取、博闻强识的精神一起进步，收获了满满的友谊。希望大家走向更广阔的天地，能够拥有热爱的事业和幸福美满的人生。

最后要感谢我的父母，求学近 20 载，父母是我生命中最坚实最伟大的后盾，他们包容我，支持我，一直默默地关注我走的每一步，无条件地为我付出所能。在走上新的人生阶段之际，我会用自己的努力来回报他们，给予他们更好的生活。

六年的北邮时光转瞬即逝，我在这一片土地上成长，变成一个更美好的自己。感谢北邮，让我踏踏实实地走过生命中最美好的时光，满怀着勇气和希望去面对人生中更多的未知和挑战。