北京大学

本科生毕业论文

题目：基于触摸屏的3D图像标注工具设计

姓　　名：张吉安

学　　号：00848160

院　　系：信息科学技术学院

专　　业：计算机

研究方向：计算机应用技术

导师姓名：王亦洲

二〇一二年五月

基于触摸屏的3D图像标注工具设计

张吉安　计算机
导师姓名：王亦洲

摘要

由于3D播放设备以及3D电影和电视的日益普及，将普通的视频转换为3D视频也成为了一个极其具有实用意义的研究课题。因为技术水平以及一些固有的难题，目前的3D视频转换仍然不可避免地要加入人工交互的操作使得生成的3D视频尽量逼近真实3D视频。但是现有已知的交互标注的工具，它们对于图像的标注部分无一例外基于传统PC，这对于提高标注效率是不利的。所以我们希望能够开发出一套基于触摸屏设备的标注工具，以提高标注工作的效率。

在毕业设计的过程中，我以已有的标注模型和图像分割算法为基础，完成了基于触摸屏的图像三维信息标注工具的设计，并且基于Android 移动平台开发了一款图像标注工具。

关键词：触摸屏，3D，人机交互，Android

An Design For 3D Image Annotation Tool Based on Touch Screen

Zhangji'an Computer Science
Directed by Prof.Wang Yizhou

Abstract

Since the popularization of 3D playback device and 3D video, converting stereoscopic video from traditional ones has become a practical topic in movie industry. Due to the technical limitations and some immanent problem, we have to do this conversion with the help of human interaction in order to make the result more close to real ones which shot with real stereoscopic camera. But for all the known convert tools now, their image annotation parts is based on traditional PC, which is harmful to raise work efficiency of labeling. So we hope to develop a annotation tool based on touch screen that benefits labeling efficiency.

I design an annotation tool based on touch screen and develop it on Android mobile device as my graduation project.

Keywords: Touch Screen, 3D, Human-Computer Interaction, Android

目录

第一章　项目介绍

1.1　视频3D转制现况及前景

3D电影技术最早产生于19世纪90年代末期，英国电影先驱威廉·弗里斯格林(William Friese-Green)发明了使用两台播放机放映3D电影的技术，这是可以考证的最早的3D电影的播放技术。而世界上第一部真正的 3D长片则是1952年的《非洲历险记》。由于技术手段不足，拍摄成本高昂等原因，3D电影始终没有摆脱在电影工业中的边缘地位。一直到进入21世纪，随着相关技术手段的完善，出现了诸如《极地特快》这样有质量的3D长片，3D视频技术才渐渐重新出现在人们的视野中。

在被称为“3D电影的新元年”的2009年，涌现了诸如《飞屋环游记》《冰河世纪3》以及堪称电影史伟大里程碑的《阿凡达》。2009年底由詹姆斯·卡梅隆(James Cameron)导演的3D电影《阿凡达》的上映则使得原先渐渐回暖的3D电影市场火爆异常，使得大家不再将3D电影视为游乐场中的一种游乐项目，而是真正将其视为电影工业技术中的一支。

而随着《阿凡达》获得巨大成功以来，各个制片公司都纷纷为最新的主打电影采用3D技术。经过不完全统计，2012年在中国上映的3D电影就有22部之多。而随着3D电影带来的热潮，各类3D相关的产品也大行其道。2010年迪士尼公司旗下的美国娱乐和ESPN 宣布成立了全球首个3D电视频道ESPN 3D，美国探索传播公司旗下的探索发现频道也在同年同索尼公司和IMAX公司联手成立了3D频道。而我国的首个3D频道也在2012年元旦开始播放节目。在播放设备方面，三星，松下，康佳等国内外公司都推出了3D平板电视，任天堂，LG和HTC等公司推出了支持裸眼3D 显示的游戏机和移动电话，英伟达公司推出了基于PC的3D立体显示技术3D Vision。在娱乐方面则出现了诸如《半条命2》,《孤岛惊魂2》等支持3D效

果的游戏。

1.2 项目背景

在我们回顾3D视频相关发展的时候可以注意到，走在前沿的相关技术往往是3D显示技术。而真正的3D视频制作技术的发展却跟不上显示设备的发展。事实上现有的3D视频拍摄的成本是十分昂贵的，《阿凡达》的制作成本高达3亿美金，而《变形金刚3》的3D版本仅仅在制作费的支出上就要高出2D版本3000万美金。所以为了应对各种需要较多3D片源的同时要求控制成本的情况(例如3D电视频道)，发展将普通视频转换为3D视频的技术就显得十分必要。

不仅如此，随着3D电影越来越受到欢迎，将原先的经典电影转换为3D电影也成了一种潮流。最成功的例子就是经典电影《泰坦尼克号》的 3D转制版，在2012年4月上映之后在全球获得了3.3亿美元的票房。而我国经典动画片《大闹天宫》的3D转制版也在同年上映。但是《3D 泰坦尼克号》巨大的成功背后却是高达1800万美金的制作成本。所以研发低制作成本的3D转制技术是3D产业发展的关键所在。

目前存在的3D转制技术可以分为全自动和半自动两种，其中全自动技术所产生的效果和真实3D视频观影感受相去甚远。半自动技术虽然转换效果较好，但是由于加入了人工交互的环节所以存在效率不高的问题。在之前开发2D到3D视频交互式转换系统的过程中就发现由于传统PC在图像标注的交互中存在的不够直观，以及频繁的鼠标操作容易给工作人员带来疲劳而导致工作效率下降的问题。

基于触摸屏的标注工具的设计和开发就是针对上述问题，力图使用触摸屏交互直观的特性解决由于传统PC交互所带来的效率低下的缺陷。希望能够通过新的交互方式提高人工标注的效率，从而提高软件整体的工作效率。

1.3 项目具体工作

在本项目中我的工作主要是设计基于触摸屏的单幅图像的标注工具及其实现。

第一，根据视频标注操作的特性结合触摸屏交互的特点，整理出视频交互标注操作的具体模型。依据所整理出的模型为标注工具设计完整的软件模型。

第二，在Android平台的移动设备上实现第一步设计出的软件模型，根据实现的结果对于该软件模型的可行性做出评估。

第二章　系统设计概述

2.1　视频标注系统整体概述

下图是一个成熟的视频标注系统的整体结构。 可以看出，在一个视频标注系

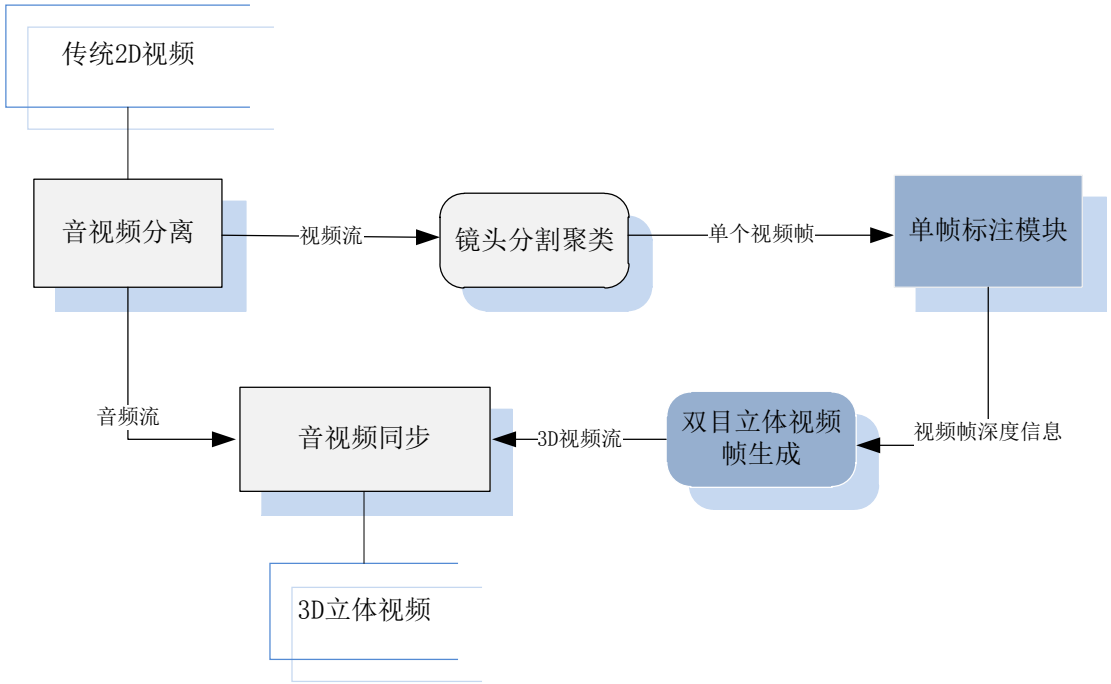

图 2.1: 系统流程

统中，已经存在完善且成熟的技术的有音视频分离和音视频同步部分，而镜头聚类分割和双目立体视频的生成也有了成熟的自动转换的技术。

所以实际上交互式的视频标注系统中的交互标注操作几乎都完全花费在单帧的标注操作上，事实上在我们已有的基于传统PC上的标注工具的使用中发现，除了视频的单帧标注之外，其余的操作可以完全地实现自动化。

2.2 视频帧交互式标注模块

单帧的交互式工具的目的是为了在单幅图像上标记出图像的深度信息，生成对应于图像的depth map。同时这也是本项目工作的重点。

目前针对视频帧的标注模块大致可以分为两个方面，分别为前景部分的深度标注和背景的深度标注。其中背景的深度估计对于估计的准确度和物体互相之间的深度的把握较前景的要求要低，而前景的深度估计则在物体的深度变化和物体之间的深度关系上和观影者的感受较为密切。也就是说，在背景的估计上我们可以使用较为粗略但是效率较高的方法，而在前景的标注上则要求尽量能够反映前景物体的深度信息以带给观影者较为真实的3D观影感受。 目前可利用的背景估

图 2.2: 自动/手动深度估计的对比

计的方法有两种，一种是利用Stage Model[1]来对于背景场景建模，另一种则是利用自动的深度估计来估计背景的深度。

从图2.2中可以看出，由于仅仅对于场景的几何结构做出了估计，Stage Model对于背景的估计效果实际上是很不好的。例如背景中一些有层次的效果在Stage Model中体现不出来。相反地，对于背景中的物体，背景深度自动估计的方法却可以估计出比较真实的效果。所以最终我们选择了以背景深度自动估计的方法来做背景深度的标注。

而同样来自于图2.2的信息也可以看出对于自动的深度估计在细致的部分上的表现效果很差，这样在观影的时候由于观众的注意力集中在前景物体上，所以微小的缺陷也容易给观众带来不适。

所以在前景的标注上我们需要采用手工标注深度的方法使得在前景部分的深度估计较为准确，从而得到更为精确的深度图，在转换成立体视频的时候更加逼近真实拍摄的立体视频。

2.3 系统其他部分介绍

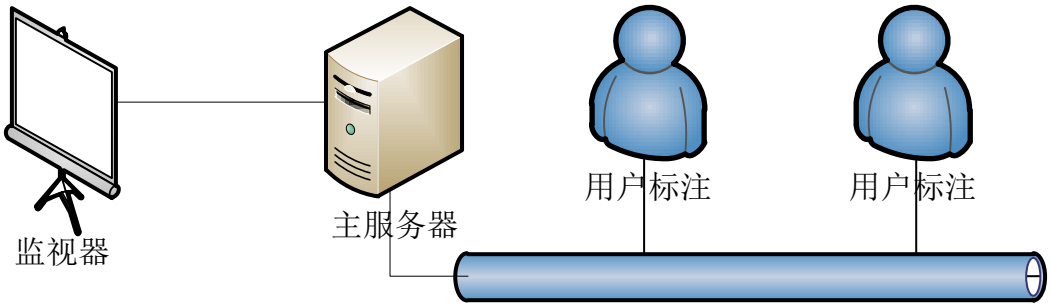

图 2.3: 系统架构

- 音视频分离与合成模块:

  目前有两种比较成熟的技术分别为微软公司的DirectShow技术。DirectShow是目前商业上成熟的视频播放软件与视频处理软件所常用的技术。另一种为FFmpeg项目。FFmpeg是一个开源的免费的跨平台音视频流方案。它属于自由软件，根据所选择应用的组件不同需要遵守LGPL 或者GPL许可证，是众多优秀开源视频处理软件所采用的方案。

在这两种方案之中Directshow是一个比较成熟的方案，优点在于创建简单的音视频处理流程

- 镜头分割和聚类模块和立体图像生成：

  这两项工作已经有成熟的可运行的工作可以借鉴。这里就不再赘述。

图2.3是我们构想中的标注系统的示意图，可以看到最终的设想架构是在传统具有高计算性能的服务器上运行音视频分离，背景深度图自动估计和立体视频生成等需要消耗大量计算量的模块。而客户端则采用本项目中设计的标注模块运行单帧的标注工作。为了提高计算资源的使用率可以将系统设计成分布式系统，多个客户端共享一个高性能服务器。

第三章　标注模型的设计

在设计标注模型的时候，我们充分考虑了触摸屏的操作特性，综合了计算方法的计算资源开销和人工操作的时间开销：

3.1　基于触摸屏的标注模型

在作为标注媒介方面，和传统的PC操作触摸屏具有以下几个优势:

首先是触摸屏的操作直观，由于图像的深度信息标注类似于在图上作画，而基于传统PC使用鼠标点击特定图片已达到标注信息的目的是非常不直观而且效率低下的。

与此同时，触摸屏的标注就要显得直接的多，可以将屏幕作为画布，在屏幕上直接点击对应点达到标注的效果。虽然面临着类似于手指标注不够准确的问题，但是这个问题可以用触控笔和屏幕的结合来解决。

其次多点触控(但是据悉苹果公司已经获得了该项技术的专利，将其应用到产品上可能带来潜在的版权风险)以及手势识别相较点击按键更为便利。

在我们之前的软件实际操作中发现过多的按键对于工作人员的效率影响很大，从而降低工具的标注效率。另一方面如果更多地使用手势来代替按键，就能够将更多的屏幕面积用于显示标注的图像，这样就能够减少由于不必要的图像缩放造成的标注之外的操作。而且使用手势来代替按钮从软件的使用上就更为简易，对于培训新的工作人员上手操作是十分有利的。

第三，触摸屏能够避免长时间操控鼠标带来的伤害

事实上在设计标注工具的时候我们需要考虑到该工具的易用性，由于在电影工业中实际的图像立体信息的标注需要工作人员长时间的专注于重复的信息标注工作。长时间操作鼠标容易给人带来疲劳的感觉。另一方面长期长时间在工作时

使用鼠标容易给工作人员的身体健康尤其是指关节带来不利的影响，甚至导致患上相关疾病。

然而触摸屏由于操作更贴进人体正常操作，所以对于减缓工作疲劳和减轻相关疾病的概率都是有利的。这从另一个方面也提高了标注工具的工作效率。

和2.2节中所述的一样，我们将整个标注模型分为了前景标注和背景标注。整体标注流程如图3.1： 可以看出，我们在单帧的前景标注中分别生成了两幅depth

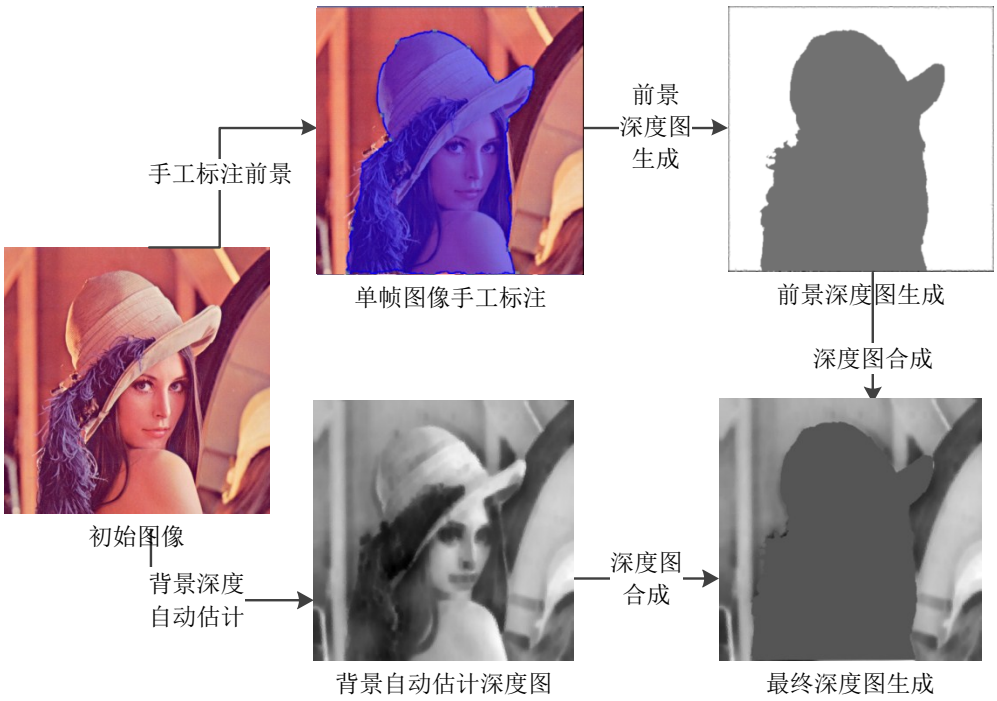

图 3.1: 单帧标注模型的设计

map，最后将其合成为最终的depth map从而达到标注的目的。事实上在这里大量的背景标注工作被交由计算机自动生成。因为背景标注的效果虽然不如手动的标注精确，但是经过我们的实践标明只要保证背景的深度估计在时间序上是连续的就不会给观影者带来不适。这主要是由于观众的注意力主要集中于深度变化较大的前景物体上。

由于背景的深度自动估计已经有了成熟稳定的算法，所以本项目的主要目的是在解决前景的标注问题。

3.2 前景标注方法

前景标注有很多方法，其中最传统的是手工标注出每一个像素点的深度值。这个方法的优点是易于实现，而且如果能保证人工标注完全准确的话，理论上可以得到ground truth。但是这个方法的缺点显而易见，那就是工作量非常大，工作效率非常低。而且如何保证帧之间的标注一致性也是一个问题。

除去逐点标注的方法，由于前景物体大多是独立于背景的一个物体，所以可以将前景图像切分出来，并且给前景图像单独赋予一个深度值的办法来将一整块的代表前景物体的像素都赋予一个相同的深度值用来代表物体的深度。这种图像切割的方法大概有两种成熟的方法可以选择。

第一种较为实用的方法是使用Graph Cuts[2]的方法。其使用方法如图3.2所示，需要先设定一个矩形框，指明前景的大致位置。并且在矩形框中人工地标注上一些属于前景和背景的位置来引导图形的分割。实际上不需要将所有前景和背景都标注出来，只需要使标注能够覆盖大致的前景和背景的范围即可

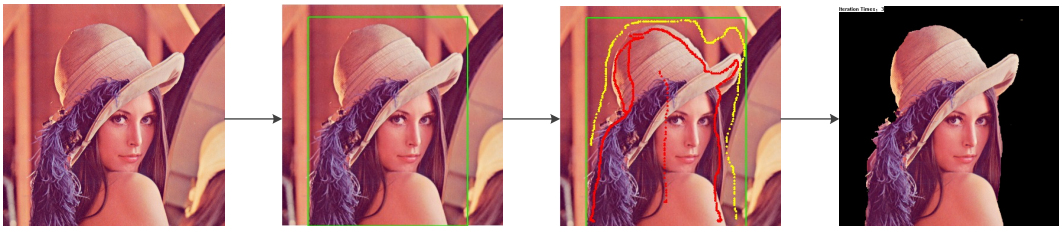

图 3.2: Graph Cuts标注流程

第二种方法是使用Intelligent Scissors[3]的方法来对图像进行分割。如图3.3所示，可以看到Intelligent Scissors类似著名图像处理软件Photoshop的磁性套索。实际上这不是一个整体的图像切分的算法，而仅仅是图像不规则边缘的切分算法。

为了使Intelligent Scissors能在我们的算法中切分出一个封闭区域，我们对于它的使用做如下的限制：在初始分割一个前景物体的时候可以从前景边缘的任意位置开始标注。而只要这个前景物体被确定了第一段边缘之后，剩下的所有分割操作就只能是在这个已有的边缘上以这条边缘划定的不规则曲线的端点为基础扩展。

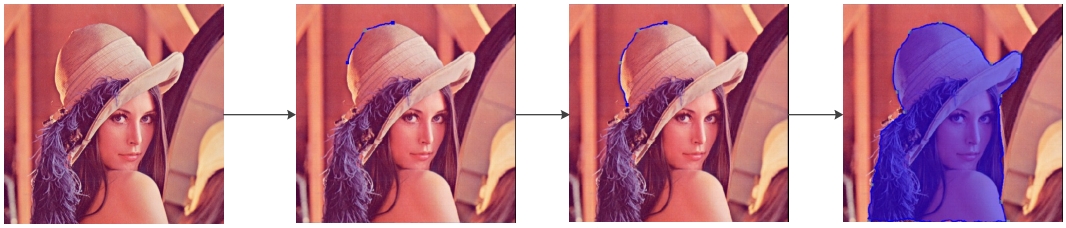

图 3.3: Intelligent Scissors标注过程

如图3.3中所显示的那样，要延伸在标注中的前景物体的边缘的方式就是从当前边缘曲线的两端开始扩展。这样就能够保证最后当用户将当前边缘曲线的两个端点连接起来的时候一定能够得到一个封闭的区域作为前景物体的标注区域。具体的标注如3.4 所示的那样，用户需要将曲线$\widehat{AB}$作为一个边缘分割出来就只需要先点击A，再点击B，则算法会自动将最贴近图像边缘的曲线画出来。

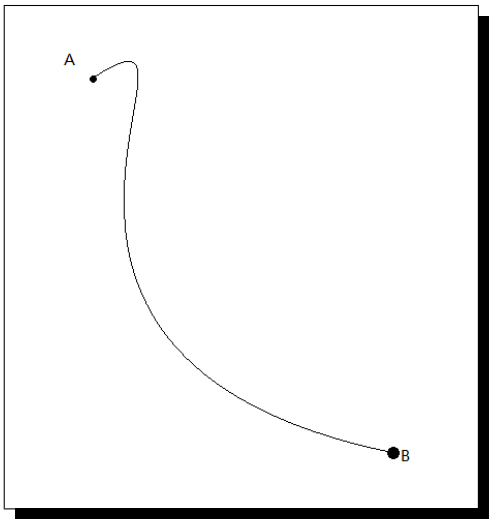

图 3.4: 切割一条曲线

基于Graph Cuts的分割和基于Intelligent Scissors的分割都能够达到我们对于前景物体的抠像的目的。这里我们将两个方法的特点做出一个对比：

基于Graph Cuts的分割可以一次性地分割出整个前景物体的轮廓，这样对于用户来说省时省力。而且分割出来的图像一定是一个封闭的图形。这个特性使得我们在编写代码的时候不需要考虑闭合轮廓的这个过程。并且我们可以一次性得到前景和背景的mask矩阵。

而基于Intelligent Scissors的分割实际上专注于解决物体的轮廓问题。这个方法的计算量较小，但是在理想的情况下来看操作的次数相较Graph Cuts。因为首先Intelligent Sissors不可能一次性将前景物体分割出来。而且由图3.4我们也可以看出，实际上每次标注的两点不能相隔太远或者用来标注如图3.5的情况：我们需要标注曲线a为边缘，但是点击A，B两点之后返回的很大概率(取决于实际图像的

情况)会是c，以及b。所以为了得到曲线a就需要手工在a上多标注一个点。实际上如果一个前景物体包含了许多类似曲线 a这样弧度很大的边缘部分，那么我们就需要标注每一个顶点才能完整地得到正确的曲线。

而且相比于Graph Cuts松散的标注要求，Intelligent Scissors完全信任用户的标注，所以需要用户标注的每个控制点都是准确的。

不过如果考虑非理想的情况就可以看出Graph Cuts的缺点：这个算法不能100%保证切割出来的前景和背景是一定准确的。或者说很有可能标注的每一个帧上都有细小的误差而需要调整。而调整部分又需要使用Intelligent Scissors算法重新计算不准确部分的边缘。

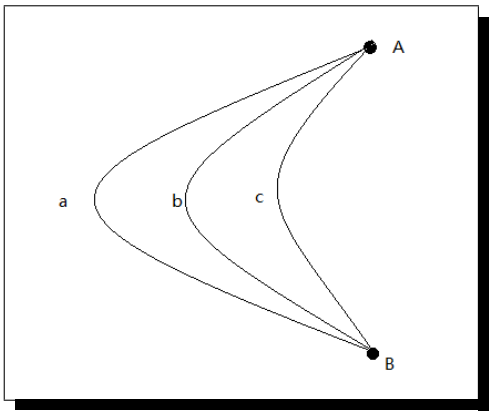

图 3.5: 多条曲线过两点的情况

实际上，我们标注的要求是得到前景物体尽可能精确的前景图形的闭合边缘，至于视频帧上每一个像素点是属于前景亦或是属于背景虽然是我们需要求解的问题，但是一旦我们得到了闭合边缘曲线，那么曲线所包含的部分自然就是前景物体了。另一个方面如果得到了前景和背景的范围根据它们相交的部分也可以得到边缘曲线。

不过由于边缘部分的前背景划分可信度是最低的，所以往往我们最能够信任的是我们不关心的前景中心部分的标注(很少有情况) 会将前景中心的像素错分成背景，我们最关心的边缘部分的划分反而是最不可信的(这会在3.3.1中详细叙述)。

基于上述的考虑，结合我们的设备选取的是触摸屏的情况，最终选择了Intelligent Scissors作为前景分割的工具

3.3 标注中使用的算法简介

在本节中将要简要介绍本章中涉及的算法。

3.3.1 Graph Cuts算法

Boykov和Jolly在研究连续泛函问题的全局优化时提出了图像分割方法Interactive graph cuts[2]。在Graph Cuts中需要人工地制定前景和背景的一些点，并且从人工标注的点中得到分割的一些基础，例如前景和背景的位置，颜色信息。

如3.6所示，构造一个s-t网络[4]，网络节点集合V的组成部分由网络上的中间节点（每个节点都对应了图像上的一个像素)和源、汇点S，T组成。其中和S相连的表示前景，和T相连的表示背景。

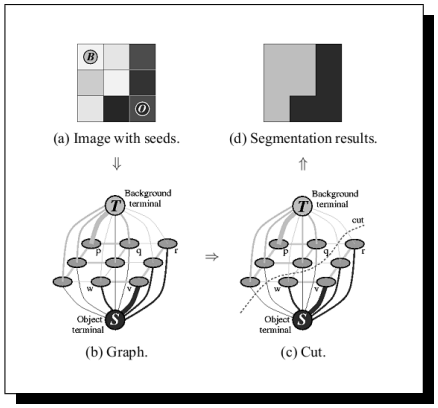

图 3.6: 一个3X3的图像分割例子

将每条边划分为两种类型：

- t-links边集：
  中间节点和源点或者汇点连接的边
- n-links边集：
  相邻像素对p,q之间连接的边。

我们需要对于每条边都赋予一个权值，以下是权值表：

边	权值	类型
$\{p,q\}$	$B_{p,q}$	$\{p,q\} \in N$
	$\lambda \cdot R_p(background)$	$p \in P, p \notin O \cup B$
$\{p,S\}$	K	$p \in O$
	0	$p \in B$
	$\lambda \cdot R_p(front)$	$p \in P, p \notin O \cup B$
$\{p,T\}$	0	$p \in O$
	K	$p \in B$

对于s-t网络的切割可以表示为：

$$A = (A_1, \cdots, A_p, \cdots, A_{|p|}) \qquad A_p \in \{"front", "background"\} \tag{3.1}$$

切割之后所有的像素点分为和源点以及与背景相连的两个部分，它们分别代表的前景和背景。定义划分的能量函数如下：

$$E(A) = \lambda \cdot R(A) + B(A) \tag{3.2}$$

$$R(A) = \sum_{p \in P} R_p(A_p) \tag{3.3}$$

$$B(A) = \sum_{\{p,q\} \in N} B_{p,q} \cdot \delta_{A_p \neq A_q} \tag{3.4}$$

可以证明能量函数的最小值就是对应于s-t网络的最小分割，这里不再多做证明。需要说明的是实际应用的方法是Graph Cuts的改进方法Grabcut[5],由Rother等人于2004年提出。主要改进在于能量函数的改进和EM思想的使用。这里不多赘述。

从上可以看出，Graph Cuts几乎无差别地估计图像上的所有点——而非专注地解决边缘问题，这和我们的问题并不是完全相符的。。正因为这个缺陷，我们在以往的实践中发现一个问题：前景的中心部分往往可以很准确地估计出来，但是在边缘处却或多或少总是有估计不准的地方。经常性地需要重新修改的边缘能够达到一半，而且多数存在于拐角等容易引人注意的地方。

在传统PC上计算资源足够的情况下我们可以选择使用Graph Cuts，牺牲一些计算资源来省却人工标注一些不容易出错的边缘。但是在移动设备上计算资源较为紧张，而操作较为便利，因此不再需要 Graph Cuts。但是需要说明的是在计算资源充足或者过剩的情况下还是可以采用这种算法来节省一些操作开销。

3.3.2 Intelligent Scissors算法

Intelligent Scissors[3]由Mortensen和Barrett于1995年提出，最终被选择作为我们的图像分割算法。

该算法如图3.4所示意的那样，用用户给出边缘曲线的起点和终点之后在图上寻找一条最短的路径作为分割出来的曲线。

参考文献

[1] V. Nedovic, A. W. M. Smeulders, A. Redert, J. M. Geusebroek. Stages As Models of Scene Geometry[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2010, **32**(9):1673–1687

[2] Yuri Y.Boykov, Marie-Pierre Jolly. Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-DImages[C]. Proceedings of Internation Conference on Computer Vision. 2001

[3] Eric N. Mortensen, William A. Barrett. Intelligent scissors for image composition[C]. Proceedings of the 22nd annual conference on Computer graphics and interactive techniques. SIGGRAPH '95, New York, NY, USA: ACM, 1995, 191–198. URL http://doi.acm.org/10.1145/218380.218442

[4] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. Introduction to Algorithms[M], third . The MIT Press, 2009

[5] C. Rother, V. Kolmogorov, A. Blake. GrabCut -Interactive Foreground Extraction using Iterated Graph Cuts[C]. Proceedings of the 31nd annual conference on Computer graphics and interactive techniques. SIGGRAPH '04, ACM, 2004, 309–314