**分 类 号 学号 M201672904**

**学校代码 10487 密级**



**硕士学位论文**

**基于计算机视觉的手势识别系统的设计和实现**

|  |  |  |
| --- | --- | --- |
| **学位申请人** | **：** | **蔡碧海** |
| **学科专业** | **：** | **计算机技术** |
| **指导教师** | **：** | **张杰** |
| **答辩日期** | **：** | **2018年5月21日** |

**A Thesis Submitted in Partial Fulfillment of the Requirements**

**for the Degree for the Master of Engineering**

**An Gesture Recognition Tool on Computer vision: Design and Implementation**

|  |  |  |
| --- | --- | --- |
| **Candidate** | **：** | **Bihai Cai** |
| **Major** | **：** | **Computer teconology** |
| **Supervisor** | **：** | **Jie Zhang** |

**Huazhong University of Science & Technology**

**Wuhan 430074, P.R.China**

**May, 2018**

**独创性声明**

本人声明所呈交的学位论文是我个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，除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期： 年 月 日

**学位论文版权使用授权书**

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密□， 在 年解密后适用本授权书。

本论文属于

不保密□。

（请在以上方框内打“√”）

学位论文作者签名： 指导教师签名：

日期： 年 月 日 日期： 年 月 日

摘 要

在人工智能和大数据技术飞速发展的今天，基于图像的识别技术也必将发挥极其重大的作用。与此同时，新的人机交互方式在也在迅猛的发展着，比如语音识别技术、手势识别技术、人脸识别技术等，这些技术使得人不必受限于遥控器，手柄，鼠标键盘等控制设备，以更自然的方式和计算机进行交互着。手势识别技术就是其中比较热门的研究方向，也必然影响着未来。一方面，新型的人机交互模式体现了人们对于人机交互中人扮演角色的思考，人往往是人机交互的主体，新的人机交互必然会朝着人性化、便捷化、自由化的方向发展。另一方面，在准确识别人的手势的基础上，也必定会影响机器人研究过程中对于人机交互的思考，准确的手势识别使得机器人能够真正看懂操纵者的“语言”。在花大精力研究遥控系统的同时，简单的手势会显得更加地便捷，并且能用简单的手势可以解决的问题，往往使得遥控器的交互过程会显得比较复杂。基于机器视觉的人体动作和手势识别是实现新一代人机交互系统所不可缺少的一项关键技术。作为一个革命性的产品，Kinect体感设备能够获取彩色图像、深度图像以及人体骨骼图像，并提供了一种全新的人机交互的方式。它能够捕捉、跟踪人体的动作、手势和声音。

在此基础之上，本课题旨在研究一种利用Kinect获取手势信息，并基于动态时间规整算法（DTW）的手势识别的交互系统，并将这种交互系统运用于实践，代替鼠标和键盘的一些基本的操作。

**关键词：**图像处理，手势识别，人机交互，动态时间规整，计算机视觉

Abstract

英文摘要，应用格式“宋体小四”

**Key words：**HUST, Thesis, Word, Template

目 录

[摘 要 I](#_Toc511198331)

[Abstract II](#_Toc511198332)

[目 录 III](#_Toc511198333)

[1 绪论 1](#_Toc511198334)

[1.1 研究背景 1](#_Toc511198335)

[1.2 手势识别概述 2](#_Toc511198336)

[1.3 论文的内容和意义 3](#_Toc511198337)

[1.4 国内外研究现状 5](#_Toc511198338)

[1.5 论文组织结构 8](#_Toc511198339)

[2 相关技术概述 9](#_Toc511198340)

[2.1 深度图像下的手势分割技术 9](#_Toc511198341)

[2.2 手势轨迹识别技术 11](#_Toc511198342)

[2.3 本章小结 13](#_Toc511198343)

[3 手势图像数据的获取和处理 14](#_Toc511198344)

[3.1 基于Kinect的需求分析 14](#_Toc511198345)

[3.2 图像数据的获取 14](#_Toc511198346)

[3.3 图像获取的模块设计 17](#_Toc511198347)

[3.4 图像数据的处理 17](#_Toc511198348)

[3.5 图像处理的模块设计 22](#_Toc511198349)

[3.6 本章小结 22](#_Toc511198350)

[4 手势识别模块的实现 23](#_Toc511198351)

[4.1 手势轨迹的获取 23](#_Toc511198352)

[4.2 基于几何特征匹配的手势识别 25](#_Toc511198353)

[4.3 DTW算法核心技术 27](#_Toc511198354)

[4.4 基于DTW算法的手势识别 27](#_Toc511198355)

[4.5 手势识别的模块设计 27](#_Toc511198356)

[4.6 本章小结 27](#_Toc511198357)

[5 系统框架搭建和测试 28](#_Toc511198358)

[5.1 软件系统的实现框架 28](#_Toc511198359)

[5.2 几种手势的应用场景 28](#_Toc511198360)

[5.3 手势识别过程分析 28](#_Toc511198361)

[5.4 本章小结 28](#_Toc511198362)

[6 总结与展望 29](#_Toc511198363)

[6.1 总结 29](#_Toc511198364)

[6.2 展望和改进 29](#_Toc511198365)

[致谢 30](#_Toc511198366)

# 绪论

## 研究背景

如今，计算机技术正在高速地发展着，在人机交互的技术方面，人和计算机的交互也越来越多样化，然而新的人机交互方式也朝着更加自然和更加便捷的方向发展着。人工智能技术的发展使得基于图像和语音的人机交互方式更加稳定和准确地能表达人与机器之间的“交流”。智能化、鲁棒性的人机交互方式是近年来学者研究的热门方向。

随着计算机互联网技术在我国的迅速普及，人们发现传统的鼠标键盘等机械输入方式，在三维空间自由度方面越来越体现其所固有的局限性，这种交互方式可能并不是对所有人都能很快上手，有一个适应学习的过程，因此更加随心所欲的人机交互方式呼之欲出。随着一些科学技术的发展，越来越多的研究更多的在关注人脸识别、语音识别、人体识别等方面对于人机交互方面的应用。

网络覆盖和网络质量是移动通信网络运营中的重要部分，网络的优化质量直接决定了移动通信网络的好坏，因此移动通信网络运营商非常重视网络优化工作。做好网络优化工作可以维持或者提高网络质量，进而在用户数和周围环境不断变化的情况下，持续提供更好的用户体验，来保证在移动通信竞争环境中保持住有力地位。

移动通信网络优化是通过对入网的设备进行数据采集和分析，找到网络中存在的问题和造成问题的原因，然后通过对网络设备参数或者通信系统参数的优化和调整，使网络性能最佳。通常网络优化包括两方面要求：解决问题和提高性能。解决问题主要是针对网络中发现的问题和故障提出有效方法加以解决和排除。提高性能是对网络中可能正常运行但是指标较差的情况，通过调整参数优化配置等方式来提高网络性能。

移动通信网络具有较强的地理空间特性，因此在进行网络优化时结合地理位置进行具体分析才有更多的现实意义。目前一些常规的网络指标分析软件与网络优化软件，大多是面向某个单一领域，没有将测试的指标数据与地理空间进行关联，更难以对网络性能指标在地理空间上做更高级的分析。而目前被广泛使用的地理信息系统（GIS）则为移动通信领域提供了一个新的视角，可以将移动通信的网络指标在空间和时间两个维度上进行结合，以直观的空间视图界面展示业务的地理空间分布和随时间变化趋势[3]。同时，通过在空间和时间上的联合分析，可以找到更多影响网络质量的细节，为最终网络优化的决策提供更加丰富的信息。

基站记录有用户详细的网络数据和用户话单记录，利用这些数据结合大数据计算进行相应的分析，可以比较实时高效地检测网络的运行状况，并及时发现问题和定位原因。云计算平台Hadoop可以用来解决大数据面临的存储和计算问题[4]，目前Hadoop主要应用于国外的Google、Facebook和国内的腾讯、阿里等互联网公司，但是在移动通信运营商中应用较少。

该论文所设计的移动通信网络覆盖性能测试工具，采用Hadoop云计算平台实现了对基站的话单数据的实时存储，并且结合相应的话单定位算法可以高效地完成对话单数据进行定位处理，得到每个话单发生的时间位置和信号质量信息。最后，结合地理信息系统，可以在地图上展示出信号质量和话务量在空间和时间维度上的分布，并通过后续的分析可以找出地图上的弱覆盖和高负载区域，这些信息对于移动通信网络优化具有重要的参考意义。

## 手势识别概述

### 静态手势识别技术

### 动态手势识别技术

这是正文。

为了保持格式的一致性，强烈建议取消word程序的“格式跟踪”功能。设置方法：点击菜单“文件|选项”，调出“选项”对话框，在“高级|编辑选项”页，取消勾选“保持格式跟踪”。

另外，建议设置文件自动保存。设置方法：点击菜单“文件|保存”，调出“选项”对话框，在“保存”页，设置自动缓存文档间隔1分钟。

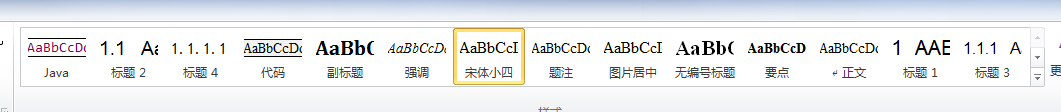


图 1‑1 应用样式列表

对于所有的标题样式直接点击图 1‑1对应样式更新，其中一级标题的样式是“标题1”，二级标题的样式是“标题2”，以此类推。并且每个标题都是自动编号的。

对于正文的样式，直接选择正文，然后点“宋体小四”，这样样式就变成正文的样式。

下面介绍其他的样式：

* Java代码、代码样式：这是java代码、伪代码的样式；
* 题注样式：这是给题注用的样式；
* 图片居中：图片使用此样式可以使图片居中显示；
* 无编号标题：这是给诸如摘要、Abstract、目录、致谢等标题使用的样式，该样式的标题没有编号。

## 论文的内容和意义

对于图片、表格、公式比较烦人的就是题注和交叉引用了。

### 论文选题的内容

放置完图片后，点“图片居中”样式，图片居中显示。

点击菜单“引用”中的“插入题注”就可以为图片添加题注了。题注有专门的应用样式“题注”。如图 1‑2，点击“引用”中的“交叉引用”可以完成对图片的引用。这样引用的好处是，如果图片的编号发生变化了，全选文字然后按F9更新域，所有的编号（包含目录）都会自动调整。



图 1‑2 这是题注

如何在一行同时添加三张图片呢？

如图 1‑3，我的做法是：

1. 插入一张2行3列的表格
2. 在第一行的表格中放入图片，在第二行的表格中放入对应的题注
3. 隐去边框

|  |  |  |
| --- | --- | --- |
| C:\Users\Jasper\Desktop\chapter03\1point.png | C:\Users\Jasper\Desktop\chapter03\2point.png | C:\Users\Jasper\Desktop\chapter03\123point.png |
| (a) 图片1 | (b) 图片2 | (c) 图片3 |

图 1‑3 一行同时添加三张图片

### 系统设计

表 1.1 表格的题注

|  |  |
| --- | --- |
| 在应用中点“插入题注”添加题注 | 如果没有“表”的标签，则需要自己创建 |
|  |  |

(1‑1)

表格如表 1.1，公式如公式(1.1)，公式直接复制过去使用，公式的编辑使用Word2010自带的公式编辑器，用它编辑公式很方便的。

### 论文选题的意义

本文的研究目的是设计并实现一个基于地理栅格的蜂窝网络覆盖性能分析的工具。本文处理的数据是某运营商在某一地区一段时间内的PCMD （Per Call Measurement Data）数据。本文所做的主要工作如下：

1. 通过Hadoop分布式处理平台对PCMD数据进行存取，生成栅格数据。PCMD数据记录的是每天通话记录的情况，包含通话时间、信号质量、基站编号、往返时延等基本信息。其中对PCMD中的基站编号和往返时延采用定位算法进行计算，可以得到每条通话的具体位置。由于PCMD 话单数据量巨大，一个月某省就能生成几十亿条话单数据，采用Hadoop 平台可以高效地对这些数据进行存取和计算，生成的栅格数据。
2. 将蜂窝网络的信号覆盖和话务负载信息以栅格图的形式进行展示。要更好地基于地理位置展示蜂窝网络的信息，需要将地理空间细分为若干个100m\*100m 的栅格。Hadoop平台生成的栅格数据包含具体位置、信号质量、话务量、时间戳和日期等信息，具体位置指栅格的ID，信号质量指某一时间段内在该栅格发生的所有通话的信号质量的平均值，话务量指某一时间段内在该栅格中发生的所有通话的数量，时间戳和日期是标记某个具体时间段。根据之前已经计算好的每个栅格的信号质量和话务量信息，根据值的大小在底图进行染色，最后生成信号覆盖和话务负载的栅格图。
3. 对蜂窝网络的信号覆盖和话务负载信息单独进行处理和分析，分别找到分析地区在该时间段内的弱覆盖区域和高负载区域。弱覆盖区域是指信号质量低于某一分析阈值的区域，高负载区域是指话务量高于某一分析阈值的区域，两个阈值都允许用户手动输入。通过数据分析，找到栅格数据中信号质量低于某一阈值的栅格，调低染色的透明度，并调高正常区域的染色透明度，在栅格图中重点突出显示弱覆盖区域。同理，突出显示高负载区域。
4. 对蜂窝网络的信号覆盖和话务负载进行联合统计与分析，统计出同时存在弱覆盖和高负载栅格的比例，并且在栅格图上重点展示此类栅格的位置。通过分析这些栅格点与基站距离之间的关系对此类需要优化的栅格点进行分类，针对基站能覆盖到的栅格点，通过调整基站参数结构、天线倾角等方式来优化；针对基站覆盖不到的栅格点，通过增建基站或者放置直放站的方式来优化。

## 国内外研究现状

### 网络优化技术和工具

随着移动网络技术不断的升级换代，移动通信网络分析和优化技术也需要不断去创新以提高网络检测的效率。传统的网络分析方式非常依靠人力，例如路测需要用专业的车辆和设备去采集相关数据，并且需要相关专家对数据做专业的分析。所以路测是比较低效的解决方法。为此，3GPP（the 3rd Generation Partnership Project）在其第9版标准[5]中已经包含了MDT（Minimization of Drive Tests），因此使用此版本网络通信标准可以显著降低成本。MDT的基本概念是用户的终端（the User Equipment, UEs）可以根据操作请求向网络报告它们的地理位置。传统的UE的测量和MDT的主要区别在于：传统的UE 是基于小区基站得到的地理位置，而MDT是基于用户终端的GPS技术得到的地理位置[6]。运营维护人员可以直接利用这些数据进行网络操作管理和优化任务。同时，研究人员也可以利用这些定位数据进行网络预测，进而为用户提供更好的网络质量。

为了建立一个准确和可依赖的覆盖地图，文献[7]提到一种从统计学借鉴来的空间插值技术Kriging[8]。 这种技术依赖于所测的数据的相关性并且可以在感兴趣的区域绘制出一个完整地图。在一些论文中应用Kriging技术[9]去做覆盖地图预测。在许多文献都研究Kriging和它的一些衍生技术来进行覆盖预测。

在文献，Galindo-Serrano等提出无线环境地图（Raido Environment Maps, REM）的方法用来解决蜂窝网络覆盖盲区检测的问题。REM对那些有地理位置的数据应用了空间差值的技术来得到真实的地理数据。这种方式可以自动鉴别覆盖盲区的数量、位置和形状。REM 可以对覆盖盲区的检测和预测效率都会有提高。

Milola提出在文献[12]提出地理位置的数据信息需要位置/环境信息。REM存储了地理位置信息、移动通信网络测量数据、环境信息以及过去的数据。在文献[13, 14]中，作者提出一种网状的REM结构，IC（Interference Cartography），这种结构中的点是方形栅格（比如像素点）。该方法的主要观点是：

1）用收集得到的带有地理位置的测量数据来预测未测量位置的测量数据；

2）测试未测量位置来提高预测的质量。

然而算法的复杂性随着测量点数数目的增加成指数增长O(N3)，N是测量点的数目）。在文献[15]中提到的FRK （Fixed Rank Kriging）是Kriging的一种变型，它的算法复杂度为O(NR2)，R是由用户定义的“固定等级”。在文献[16, 17]，该算法被用于覆盖预测。在仿真和实际测试的性能评估中，FRK均被证明在算法复杂度和预测准确度之间实现了非常好的平衡。

但是之前的这些工作大多是建立在移动终端可以准确定位的理想情况下的。然而，文献[18]中指出使用GPS定位的误差在5m到30m范围内，而在文献[19]中说明基于无线网络度量的定位技术误差在50m到300m。这都说明定位的不确定性降低了覆盖预测的准确性。

在文献[20]中，A. Palaios等人提出通过采用多种测量数据来提高定位的准确度和减少定位的误差。这种定位的方式是通过适当地组合不同传感器测到的结果，这种方法虽然能够得到不多的结果，但是显然不适用于用MDT特性测量的情况。在文献[21]中，作者Braham等人提出了通过扩展FRK算法来处理定位不准确的问题。在预测和校准模型中，在FRK模型中定位的不准确性影响了函数的均值和协方差。该文献的主要贡献是1）通过在模型中引入定位的不确定性，作者比较和研究了最佳线性无偏预测值和条件期望预测；2）引入了SAEM（Stochastic Approximation EM）算法。SAEM结合了随机EM和Gibbs抽样程序来处理大量的计算[22]。Gibbs 算法用并行处理的方法解决了定位概率密度抽样的问题。

至于网络优化工具，国际上的优化系统厂商在全球的无线网络优化市场中占据着较大的市场份额，其中爱立信公司开发的Tems优化系统应该最为广泛[23]。在国内，例如华为、中兴等公司也在开发相应的网络优化软件，但是开发的优化软件虽然在最近几年已经取得了长足的进步，但是还是存在一定的问题，首先在网络优化的实践过程中仍然需要人工参与，缺少自动化操作；其次是采用较为传统的数据存储和计算方法，对与海量的通信数据的处理效率不高；最后，仅仅是针对某种特定类型的数据提供处理，优化系统通用性有待提高。

### 地理信息系统

地理信息系统是在计算机软硬件的支持下，采集、管理、检索、分析和描述与地球表面空间位置相关的数据计算机系统[24]。它起步于60 年代，是一门集合了计算机科学、地理学、空间科学、环境科学和遥感测绘学的学科，它采用的基本技术是地理空间数据库技术、地图可视化技术和地理空间分析技术[25]。

近年来，随着科技进步和社会发展，地理信息系统在各行各业已经得到了广泛的应用。在移动通信服务领域，基于GIS的移动空间定位服务已经被绝大多数用户所使用，例如用户使用百度地图APP可以完成定位、路径规划和导航等功能，结合互联网数据，用户通过百度地图还可以找到附近的餐馆、银行。在通信网络的建设和优化方面，地理信息系统也有很多用武之地。在文献[26]中，作者提到使用GIS中的数字高程模型（Digital Elevation Model，DEM）以及其他相关分析方法在多山区域找到最佳建站地址；同时可以借助GIS 的空间数据分析的特点，结合移动通信网络参数和信号的传播模型，可以绘制出信号覆盖的预测图。在文献[27]中，作者通过结合现有的无线电波传播模型和基站辐射理论，提出并实现一种基于GIS的基站选址方案，该方案能够在基站选址时反映基站对于小区、学校和医院的电磁辐射情况，为监管部门在基站审批时提供一个可参考的直观化依据。在文献[28]中，作者提到使用GeoDatabse技术来应对GIS的海量数据存储的问题，并且GeoDatabase还支持多用户并发访问、版本管理和数据动态更新等功能，比传统的文件形式存储和空间数据的表示方法有巨大的优势。

按照空间数据的组织形式，可以将地理信息系统分为两种类型，一种是基于栅格的，另一种是基于矢量的。基于栅格的GIS以栅格元为地理特征的最小单元，并用同一个数据结构存储地理特征的空间位置信息和属性信息；而基于矢量的GIS是以点和线组成的，首先由点构成地理特征的边界，然后由点和这些点连接成的线共同组成对于地理特征的描述，并且在通用的数据库中存储其属性信息，通过数标建立空间位置信息与属性信息的联系。

本论文所研究的工具是基于栅格的GIS的应用，栅格图是由排列整齐的栅格点组成的，每个栅格点都有自己对应的空间位置信息和属性信息。比如栅格图是由a\*b个栅格组成，每个栅格用(x,y)坐标来标示，然后每个栅格的属性值可以通过颜色或者灰度值值进行表示。

## 论文组织结构

本论文由六个章节组成，其组织结构如下所述：

第一章介绍的是本文的的研究背景和意义，介绍国内外有关网络优化常采用的分析方法和工具，之后又介绍了GIS在移动通信网络中的应用现状，并介绍论文的主要解决的问题，最后给出了论文组织结构。

第二章介绍的是网络测量的相关参数和有关话单定位的算法，为之后的章节做铺垫。

第三章首先提出的该分析工具的需求分析和系统框架，然后依次谈到数据获取的流程和各个模块的设计工作，并详细说明了每个模块设置的意义、具体要解决的问题以及具体的执行流程。

第四章首先给出了该工具的交互流程设计，之后展开谈到每个模块的具体实现。对重点环节给出了伪代码和效果图，并且重点阐述了基于地理的覆盖性能模块对于II类目标栅格聚类的具体实现思路。

第五章是对本文所设计和实现的工具进行测试，首先分别对信号覆盖和话务负载模块单独测试，之后再联合测试，并对每一个测试的结果进行结果展示和分析，对最终的测试结果进行评估。该章节验证了本工具的各功能模块达到了预期效果。

第六章对本文进行了总结和展望，首先总结了本文的工作成果，最后客观地分析了本文所实现的蜂窝网络优化方法的不足，同时给出了未来可以改进的方向。

# 相关技术概述

## 深度图像下的手势分割技术

在移动通信系统中，衡量一个网络好坏最重要的要素是覆盖、容量、质量、频谱利用率和传输效率[1]。

### 手掌分割算法

无线网络覆盖是指在一定区域内，无线信号的强度和质量能够为用户提供移动通信服务，称该地区存在无线网络覆盖。移动通信系统的覆盖率和覆盖质量是衡量移动通信网网络质量和服务质量的重要指标[31]。在网络初期的建设和后期的维护期间，网络运营者通过测试、数据采集和分析等手段，来检验网络是否满足覆盖要求。如果不能达到要求，则需要对网络进行覆盖方面的优化。判断移动通信网络是否在某一地点形成覆盖，需要依据一定的测量参数。

对于GSM系统，判断是否存在网络覆盖只需要测试一个参数，即移动台接收电平（Rx\_Power）。一般当Rx\_Power-90dBm视为该地点存在GSM网络覆盖。在CDMA 系统中，与网络覆盖与导频强度（Ec/Io）有关，一般要求Ec/Io-12dB且Rx\_Power-90dBm 且Tx\_Power15dBm，只要这三个条件全部满足，才视为该点存在CDMA 网络的覆盖[32]。

**Rx\_Power：**在GSM系统中，Rx\_Power表示在工作频点上接收的电平强度，该参数直接反映了GSM网络覆盖的水平[33, 34]。CDMA 系统中，Rx\_Power表示在整个1.2288MHz带宽上总的接收功率[35]。该接收功率不仅包括服务小区的信号功率，也包括其他小区的信号功率、接收机热噪声及外界干扰。

**Ec/Io：**Yang在文献[36]介绍Ec/Io指每码片辛哈能量与总功率密度之比。在CDMA 系统中，Ec/Io反映了信号的干扰水平。移动台或者基站均要求信号的Ec/Io必须超过一定阈值才能正确解调。Ec/Io在-10dB 以上时，网络质量可以保持非常良好的水平；当Ec/Io-15dB时，则移动台无法接入网络。一般将Ec/Io-12dB作为满足网络的覆盖的标准。在CDMA系统中，Ec/Io和Rx\_Power 是评价网络前向覆盖能力的指标。

**Tx\_Power：**在CDMA系统中，移动台发射电平Tx\_Power反映网络的反向覆盖能力。CDMA系统对方向链路采用快速功率控制。在反向传播链路不好的情况下，会迅速提高移动台的发射功率，因此移动台的发射功率Tx\_Power可以衡量反向覆盖的水平。当Tx\_Power15dBm时，可认为该地点达到反向覆盖的要求[37]。

### 手掌轮廓获取算法

对于一个移动通信系统而言，除保障覆盖和质量要求外，容量的保障也至关重要。容量优化是网络优化工作中必不可少的一部分，移动给用户数量的增长、用户通信行为的变化、新业务的使用及设备故障灯，都会引起网络的可用容量降低，导致用户尝试接入网络时可用资源分配。因此，为保证移动网络高质量地运行，必须实时监控网络的容量变化，及时发现网络容量不足的问题，并采取相应增扩容量的措施，以满足用户通信的要求。

同时，作为移动通信运营商，在满足用户的话务需求的同时，也要充分合理利用既有网络设备和频谱资源，实现资源利用率最大化，从而达到提供优质服务、节省投资的目的。这也是网络优化的目标和原则。

无线网络的容量体现了移动通信网络提供服务的能力。移动通信系统中，使用无线电波作为信息的传输载体，完成用户终端与基站之间的信息传送。由于无线频谱的资源限制，使得无线网络容量的瓶颈不同于固定网络。固定网络的容量“瓶颈”主要在于交换机的交换能力，而移动通信网络的容量瓶颈则取决于无线频谱的宽度、频率复用方式和调制方式等。

在通信系统中，无论是固定通信系统还是移动通信系统，都用话务量来反映通信业务量的大小，也用来反映网络容量的大小和通信设备的负荷。

在文献[38, 39]提到，话务量的单位是爱尔兰（Erlang，简写为Erl），是为了纪念话务理论的创始人，丹麦学者A. K. Erlang 而命名的。爱尔兰定义为单位时间内信道被占用的时长。如果一个信道在1小时内被全部连续占用，称此时的话务量为1Erl；如果一个信道在1小时内被占用30分钟，称此时的话务量为0.5Erl。

话务量的计算方法为：单位时间内发生的呼叫次数和每次呼叫所占用的时间的乘积[40]。话务量一般用A表示，即

(2‑1)

其中，为单位时间内的呼叫次数，为每次呼叫的保持时间。

从话务量的公式可以看出，话务量受两个因素的影响：一个是单位时间内的呼叫次数，用户越多或者呼叫越频繁，话务量就越高；另一个是呼叫保持时间，用户的通话保持时间越长，话务量就会越高。

对于有多个用户的系统，系统的总话务量等于所有用户话务量之和：

(2‑2)

其中，中系统的总话务量，为系统中总的用户数量，为平均每个用户的话务量。

值得注意的是，通信网络中各个时段的话务量往往并不是平均分配的，话务量最高的一个小时称为忙时。忙时这一个小时的话务量称为忙时话务量。忙时话务量与全天话务量之比，称为忙时集中系数。忙时集中系数一般在10%~15%之间。

因此，忙时话务量可由下式计算，即

(2‑3)

式中，为全天话务量，为忙时集中系数。

在进行网络规划和优化时，一般以满足网络忙时的话务量需求作为规划和优化的目标。

### 手指尖点检测的算法

## 手势轨迹识别技术

在PCMD话单数据中提取用于话单定位的数据，针对每条话单连接基站数目的不同分别采用不同的定位算法。

**COO(Cell of Origin)**定位算法的基本原理是根据移动终端距离基站的距离和扇区的编号来确定移动终端的位置[41]。该论文中使用COO 定位算法对只连接一个基站的话单进行定位。



图 2‑1单点定位示意图

如图 2‑1，P1为小区位置，是小区方位角，是小区半波瓣角60，为小区传播时延换算的接入距离，介于[-,+]的随机数。接入位置以P1为原点，以 方向，半径为r处的位置。

**AOA(Angle of Arrival)**定位算法是基于方向角的两基站定位算法。文献[1]提到，其基本原理是利用移动终端到达两个基站的距离和所在的扇区的编号来计算其具体位置。如图 2‑2：



图 2‑2两点定位示意图

* 两圆相离(d>r1+r2)：d为两小区之间直线距离，以P1为原点，P1、P2连线方向，r1为接入距离；
* 两圆相含(d+r1<r2)：d为两小区之间直线距离，以P1为原点，P1、P2连线方向，r1为接入距离；
* 两圆相交(d<r1+r2)：求出两交点G\_1、G\_2；小区P1、P2的方向角为 和；G1和P1、P2 连线的方位角分别为、；G2和P1、P2连线的方位角分别为、；取角度偏差小的点，即如果，则接入位置为G1，否则为G2。

**TOA(Time of Arrival)/TDOA(Time Difference of Arrival)**定位算法基本思想是找到移动终端距离三个移动终端的交叉点即为移动终端的位置，该方法要求移动终端和基站的时间精确同步。为了测量移动终端的发射信号的到达时间,需要在每个基站处设置一个位置测量单元，为了避免定位点的模糊性，该论文中使用TOA/TDOA 定位算法对连接至少三个基站的话单进行定位，如下图 2‑3。所示：



图 2‑3三点及三点以上定位示意图

* 两圆相离、相含：处理方法同AOA，不再考虑第三点。
* 两圆相交：求出P1、P2圆的交点G1 和G2;求出P3到G1、G2的距离分别为d31 和d32；用P3的接入距离r3进行选取，如果|d31-r3|>|d{32-r3|, 则接入距离为G2，否则为G1。

## 本章小结

本章介绍了网络覆盖分析最新的一些技术以及网络覆盖和网络容量经常用到的一些测量参数，这些参数中部分会在之后的章节中出现。在这之后本章又介绍了地理信息系统的一些概念，重点讲解了下地理栅格产生的背景。最后，本文概要地介绍了下话单的定位算法。

# 手势图像数据的获取和处理

## 基于Kinect的需求分析

Kinect设备很方便的提供了深度图像、彩色图像、红外图像以及人体骨骼数据，这些数据使得开发人员对于人体骨架的25个关节点的坐标信息在摄像机坐标系中的获取和追踪变得十分方便。在微软的Kinect for Windows SDK v2中提供的接口函数中，人的手掌信息是由人体骨骼在手掌区域的3个关节点（手掌心点、手指尖点、手腕点）来确定的，然而这些信息并不能准确的提供手掌和手指的轮廓信息以及指尖点信息，因此本文中对于手势轨迹获取，是使用Kinect获取骨骼坐标点以及深度图像的相关数据，然后进行了一些必要的处理，用以更加方便和准确的获取手掌心点的坐标信息。

本文基于Kinect的骨骼图像和深度图像获取手势信息的流程如图 3‑1所示：首先通过骨骼信息中掌心点在Kinect摄像头三维空间坐标系中的位置，确定其在深度图像中的位置，基于这些信息利用深度阈值法获取手掌区域，然后对获取到的数据再一次结合深度图像进行手掌轮廓的提取，最后基于获取到的轮廓信息进一步分析手掌中心点和手指尖点。

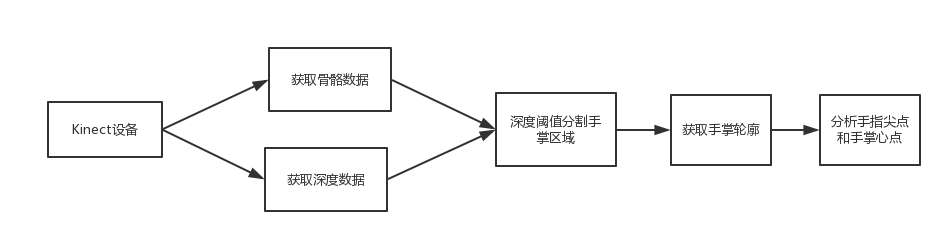


图 3‑1 Kinet图像模块的流程图

## 图像数据的获取

### 利用Kinect获取人体骨骼数据

与2011年微软推出的Kinect 1代相比，Kinect2在数据稳定性上有着非常显著的提升，其中一点就是Kinect对于骨骼数据的获取。Kinect采集了人体25个关节点，如图 3‑2所示：

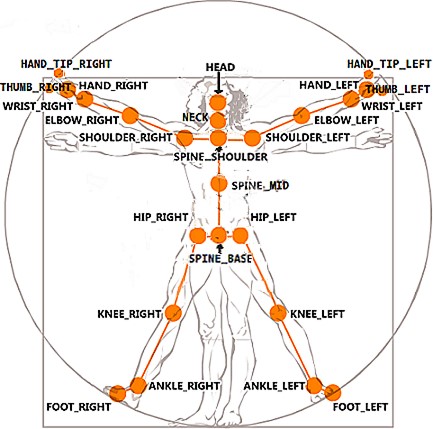


图 3‑2 Kinect2 中获取并追踪的25个关节点信息

并对这些人体的关节点进行追踪。在微软提供的Kinect for Windows SDK v2中，关节点信息是存储在一个叫Jonit的结构体中的，如图 3‑2所示：

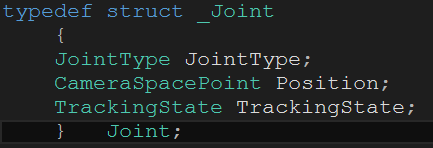


图 3‑3 Kinect SDK中定义的关节点结构体

在这个结构体中存储了三个信息：关节点类型（JointType）、关节点的摄像头空间坐标点（Position）和追踪状态（TrackingState）；另外，Kinect可以最多一次性跟踪6个人体的数据，存放在一个叫IBody的类对象的数组中，从每一个IBody对象出发可以获取到与每个人体相关的骨骼关节点。

在论文中，提取人体骨骼关节点的主要步骤是：

第一步：初始化Kinect传感器（IkinectSensor）结构体指针，打开Kinect设备，做好一系列的准备工作；

第二步：通过获取到的IkinectSensor结构体指针循环获取身体帧数据，提取出身体帧数据源（IBodyFrameSource），并打开身体帧读取器（IBodyReader）；

第三步：根据获取的身体帧读取器获取到各个关节点信息，存放到一个Joint结构体数组中，获取人体的25个关节点；

第四步：读取帧数据并重复第二步和第三步，直到最后程序关闭，关闭Kinect传感器，并释放相关指针所占据的内存空间。

### 利用Kinect获取深度图像数据

Kinect设备的另外一个特性就是可以很方便的物体在三维空间中的深度图像：相较于Kinect1代提供的320\*240的深度图像而言，Kinect2能提供512 \* 424范围的深度图像。由于深度图像的坐标系是一个x轴正方向向右，y轴正方向向下的坐标系，和OpenCV的坐标系统能够十分契合，可以很便捷地在OpenCV将获取到的深度图像数据用窗口显示出来，以便下一步的研究，因此本论文中的设计是将Kinect获取的骨骼数据和三维空间中的深度图数据结合起来进行处理的。

在Kinect for Windwos SDK v2提供的API函数中，获取Kinect中获取深度图像数据和获取人体骨骼数据是相似的，所不同的是，深度图像获取的数据是存储在一个大小为512\*424的一维2字节（16位）数组中，在这个数组中，每一个16位数据代表深度图中对应位置与摄像头之间的距离，那么这个数组中下标参数与对应下标的16位数据值关系是：

设获取的深度图数据用一个数组UINT16[512\*424] depthArray来表示，x表示深度图中的x坐标值，y表示深度图中的y坐标值，那么数组的下标参数为index = y\*424 + x，且depthArray[index]中存储的就是深度图中坐标(x, y)处与Kinect摄像头之间的距离。

获取深度图数据的步骤如下：

第一步：初始化Kinect传感器（IkinectSensor）结构体指针，打开Kinect设备，做好一系列的准备工作；

第二步：通过获取到的IkinectSensor结构体指针循环获取深度帧数据，提取出深度帧数据源（IDepthFrameSource），并打开深度帧读取器（IDepthReader）；

第三步：根据获取的身体帧读取器获取到各个关节点信息，存放到一个512\*424大小的一维2字节（16位）数组中，获取当前时间内的深度数据帧；

第四步：读取帧数据并重复第二步和第三步，直到最后程序关闭，关闭Kinect传感器，并释放相关指针所占据的内存空间。

## 图像获取的模块设计

本论文研究的是基于视觉图像的手势识别系统，图像获取是基于Kinect的摄像头，在搭配Kinect for Windows SDK v2的基础上，对Kinect摄像头中人体骨骼帧数据和深度帧数据进行了获取，将获取到的两种数据结合起来，计算出稳定的手掌信息，并对获取的手掌信息进行实时的跟踪。因此比较重要的一点就是图像获取的模块设计了。在本文中，获取图像的流程是基于Kinect SDK的提供的接口函数，对于接口函数中提供的多种数据源进行了整合，主要获取关键的数据，对数据进行了封装。

## 图像数据的处理

### 数据封装

通过微软提供的Kinect SDK能十分方便的获取到人体骨骼数据和深度图数据，在此基础上，本文对获取到的数据进行了封装，以方便后面进一步的研究；首先，将获取到的深度图数据存储到一个二维数组中去，再获取到人体骨骼流数据，提取手掌心点的坐标值，然后将掌心点坐标转换到深度图的坐标系中去，将两者结合起来考虑，这样做的目的是使得z方向上的数据值更加准确。

从程序设计的角度来讲，本文中设计了两个主要的类来存储从Kinect获取到的三维坐标点：HandPoint类和Hand类，HandPoint类；Hand类保存了与手掌相关的信息：（1）手区域中一些关键点的坐标值，这些点有包含手区域矩形的左上角点和右下角点信息、手掌心点、手腕点、指尖点；（2）手掌状态：从Kinect可以获取到手的五种状态，其中包括手掌张开（HandState\_Open）、手掌闭合（HandState\_Close，即握拳）、手指套索状态（HandState\_Lasso，即伸直食指和中指的状态）、未知状态（HandState\_Unknown）、未跟踪（HandState\_NotTracked）（3）手掌轮廓点集合：保存了从手掌区域提取出来的手掌轮廓上的点。

### 手掌模块的提取

对于Kinect获取到的深度数据和骨骼数据，要将手掌区域提取出来才能进一步进行手势识别，因此第一步就是提取手掌区域：由于Kinect对二维图像中手掌区域有优势的地方就是它可以提供二维图像无法提供的深度信息，且二维摄像头往往需要双目摄像头才能提供三维空间上的深度信息，而Kinect的摄像头本身就能获取红外信息，所以Kinect的摄像头能够非常便利的获取骨骼关节点的三维坐标信息，并且基于此获取到的数据是不受光照条件影响的，亦即在黑暗条件下也能准确的获取这些骨骼关节的深度信息，因而在第一步中获取手掌区域要利用Kinect提供的深度图数据，使用深度数据阈值方法提取出手掌区域，但是它也存在局限，其中比较重要的一点就是这样获取手掌区域只能利用Kinect提供的SDK来获取，而且目前Kinect在Linux和Mac OS平台上没有提供有效的驱动程序，所以平台会被限制在Windows操作系统上。

那么对于Kinect中获取手掌模块的步骤要分为三部分：获取手掌区域、获取手掌轮廓、获取手掌心点。

#### 获取手掌区域的方法

获取手掌区域的方法主要是深度阈值法：根据骨骼点能准确确定掌心点所在位置的三维坐标点，假设该点的位置是Pcenter(x, y, z)，那么根据x和y的值可以在Kinect获取的424\*512分辨率的深度图数据depth[424\*512]数组中找到相应的点，这个转换公式(3.1)如下：

(3‑1)

假设设定的阈值为k，那么选取深度图数据中深度值大小在范围[HandDepth – k, HandDepth + k]之间，则认为深度图数据中数值在这个范围内的深度值是属于手掌区域的，这是因为手掌区域的数据往往在深度值上和掌心点比较接近，而手掌本身往往是处于身体其他部位的前方，和手掌不在同一深度范围内，因此可以用这个深度平面内的数值来反馈手掌区域。其算法描述如下：

（1）定义一个布尔类型的二维数组HandArea[424][512]，初始化并全部赋值为FALSE；

（2） 获取Kinect中骨骼帧数据中的手掌心点HandCenter点，转换到深度图坐标系中去，获取在深度图坐标系中HandCenter点的深度值CenterDepth，即令CenterDepth = depth[]；

（3）设置两个下标值i和j，遍历整个深度图数据的数组，取出depth[j \* 424 + i]的值，赋值给nDepth, 即令nDepth = depth[j \* 424 + i], 判断nDepth是否在

集合内，若nDepth的值在该集合中，则令HandArea[i][j] = TRUE, 否则令HandArea[i][j]为FALSE。

这样二维数组HandArea中值为TRUE与深度图对应的位置就是提取出来的手的区域。

#### 获取手掌轮廓的方法

从前一步中提取出来手掌区域的二维数组后，下一步就是要根据提取到的手掌区域数据进行手掌轮廓提取，提取手掌的思路是根据轮廓点附近的8个点的特征值变化，判别在手掌区域中哪一些点事属于手掌中的轮廓点，哪些是属于手掌中的非轮廓点。

根据上面获取手掌区域的二维布尔矩阵HandArea可知，矩阵中的点，可以分为三大类：第一类就是手掌区域内的点，第二类是在手掌轮廓上的点， 第三类是在手掌区域外的点。这三类点分别有一下特征如下：

1. 若HandArea[i][j] = FALSE，则深度数据图中坐标点(i, j)对应的点在手掌区域外部；
2. 若HandArea[i][j] = TRUE, 且在坐标点(i, j)周围的8个点(i - 1, j - 1), (i, j - 1), (i + 1, j - 1), (i - 1, j), (i + 1, j), (i - 1, j + 1), (i, j + 1), (i + 1, j + 1)中，存在一个点(x, y ) ∈{(x,y)|, 且(x,y)≠(i,j)}，使得HandArea[x][y] = FALSE，则深度数据图中坐标点(i, j)对应的点在手掌轮廓上；
3. 若HandArea[i][j] = TRUE, 且在坐标点(i, j)周围的8个点(i - 1, j - 1), (i, j - 1), (i + 1, j - 1), (i - 1, j), (i + 1, j), (i - 1, j + 1), (i, j + 1), (i + 1, j + 1)中，对于所有的点(x,y) ∈{(x,y)|, 且(x,y)≠(i,j)}，使得HandArea[x][y] = TRUE，则深度数据图中坐标点(i, j)对应的点在手掌区域内；

算法实现步骤如下：

1. 初始化一个布尔矩阵HandOutline, 其大小为424\*512，将HandArea矩阵中的所有数据都复制到HandOutline中去；
2. 设定遍历下标i和j，i的值从1到424变化，j的值从1到512变化，获取HandArea[i][j]的值；
3. 声明一个字节的变量value，并初始化为0x00(二进制数值为00000000)，若HandArea[i][j] = FALSE，保持value的值为0x00不变；
4. 若HandArea[i][j] = TRUE，依次获取HandArea[i-1][j-1]、HandArea[i-1][j]、HandArea[i-1][j+1]、HandArea[i][j-1]、HandArea[i][j+1]、HandArea[i+1][j-1]、HandArea[i+1][j]、HandArea[i+1][j+1]得值，若获取的值为TRUE，则计算位运算 value = value << 1 | 0x01，否则令计算位运算 value = value << 1 | 0x00；如此经过8次运算，就可以将坐标点为(i,j)位置的左上（↖）、上（↑）、右上（↗）、左（←）、右（→）、左下（↙）、下（↓）、右下（↘）这8个方向的临近点所存储的布尔值转换到value的第7~0位上，并且该位置上位1表示对应方向的临近点在HandArea中为TRUE，0表示对应方向上的临近点在HandArea中为FALSE；
5. 根据第四步计算的这八个方向上临近点在HandArea中的布尔值，结合上文介绍的算法，比较容易知道，当value的值为0xFF（对应二进制数值为11111111）的时候，表示坐标(i, j)在深度图数据中对应的点是在手掌内部区域，则将HandOutline[i][j]设定为FALSE，而当value的第7~0位上存在任意一位位1的情况的时候，表示坐标(i, j)在深度图数据中对应的点是在手掌的轮廓上，则将HandOutline[i][j]设定为TRUE。

根据以上算法，可以获取到手掌轮廓上的点信息，并将这些信息存储到一个二维布尔矩阵，同时用一个数组points来记录这些轮廓上的点信息。

#### 获取手掌心点的方法

在获取了手掌轮廓之后，接下来比较关键的一步就是获取手掌心点，。本文采取的手掌心点的检测方法是根据手掌的几何特征，计算轮廓上的点到手中心点的中心点，根据几何知识可以知道，假设平面上有n个点：P­1(x1, y1), P2(x2, y2), … , Pn(xn, yn)，这些点的中心坐标设为C(xc, yc)，则中心点计算算法如公式(3‑2)所示

(‑2)

在上一步中，将获取到的轮廓点的数据都放在一个数组中了，用points来表示，即points[i].x和points[i].y分别表示公式(3‑2)中xi和yi，但是在处理中心点之前应该剔除四只手指对于获取中心点的干扰，以保障获取到的手掌心点的坐标点不会偏上，因此对于手掌轮廓分析，可以大致剔除四只手指的部分，其步骤如下：

1. 设置遍历下标i和j，在轮廓点集合中计算x方向和y方向上的最大值和最小值，设获取到的结果分别是xmin , ymin , xmax , ymax 那么对于手掌轮廓所在的区域可以用一个矩形区域包含之，这个矩形的左上角坐标是(xmin, ymin)，右下角坐标是(xmax, ymax)；
2. 双重循环：在下标i从ymin遍历到ymax的情况下，将下标j从xmin遍历到xmax，j遍历的一次称为一趟，在这一趟的遍历过程中，记录取得HandArea[i][j]中布尔值从TRUE突变到FALSE的次数，那么按照这种遍历方式取得的变化次数依次会是2、6、8、10、4、2，如图 3‑4 (a) 所示:

|  |  |
| --- | --- |
|  |  |
| (‑a) 布尔值变化次数 | (‑b) 获取中心点计算区域 |

图 3‑4 从上到下遍历获取手掌轮廓点剔除四只手指示意图

那么取得第一次出现变化次数为4的情况，就可以确定一个y轴方向上的约束值Ystrict，使得Ystrict之上的图像大略是四只手指所在的位置，Ystrict之下的图像就是需要考虑的，可以用来计算中心点的区域，如图 3‑4(b)中蓝色框所标识的区域

进行上面的处理之后，将蓝色区域中表示的位置在HandOutline矩阵中找到对应的坐标数据，根据公式(3‑2)就可以计算出手掌的中心点，从而进行下一步的研究工作。

### 手掌模块的显示

图像模块中对手掌数据进行了相应处理：对获取图像数据进行手掌分割，获取了手掌轮廓和手掌心点，对于获取到的手势数据，本文调用openCV提供的图形显示接口，对采集的数据进行处理，在窗口中进行显示

## 图像处理的模块设计

模式设计

## 本章小结

本章首先给出了手势识别系统的需求分析，详细说明了Kinect提供的几种数据，以及课题需要的内容。其次介绍了从图像数据的获取这个方面说明了获取整个手掌图像模块数据的实现过程，并介绍了人体骨骼数据和深度数据结合的模块设计。接下来介绍了本文中图像数据的处理。最后对图像模块的显示进行详细说明分析。

首先详细叙述了图像模块的程序设计过程，该模块作为本系统的核心功能之一，能够获取用户输入的手势信息，本章介绍图像模块方法的设计思想，为后续实现章节提供指导。

# 手势识别模块的实现

在获取手掌数据后，至关重要的部分就是手势识别模块的实现了。本文在基于Kinect获取的图像基础之上，对动态手势识别进行了实现，搭建了手势库。手势识别模块主要分为手势轨迹处理、DTW算法实现、手势库搭建三个步骤。本章主要从算法实现角度来介绍手势识别模块的实现。

## 手势轨迹的获取

手势轨迹是在一段时间内手掌坐标信息的集合，它有两个明显的特征：一、时效性，手势的轨迹是在一定时间内确定的，所以手势轨迹必定有个开始时段和结束时段；二、关联性，手势的轨迹是坐标点的集合，集合中点不是孤立的，往往点与点之间有着关联。

### 坐标转换

从获取到的帧数据手掌心点信息中，可以比较精准地获取到掌心点在Kinect深度图像这个二维平面上的坐标信息。考虑到坐标系的本身的约束，获取到的坐标点往往只是在当前坐标系中才有实际意义，因此获取手掌心点坐标点后还要对这些数据进行坐标系转化，以提供给手势系统的坐标系使用。对于坐标系转换，有两种可行的方法：

第一种方法是等比例映射法：对于平面上的两个坐标系Oxy和Ox’y’，用点(x, y)以及点(x’,y’)分别来表示这个两个坐标系中两个可以互相转化的点，设由x转换到x’的比例系数是k1，由y转换到y’的比例系数是k2，那么这两个点之间的转换算法如公式(4‑1)所示：

(‑1)

对于一般的可视化的坐标，都是以屏幕向右的方向为x轴的正方向，屏幕向下的放下为y轴的正方向，坐标值都为正值，可视的范围是整个屏幕或者整个窗口的大小，所以可视点的坐标值会规范到一个矩形之中。假设待转换的两个可视化的坐标系的矩形分别为frame1和frame2，那么如果由frame1表示的坐标系映射到frame2表示的坐标系，k1 = frame2.width / frame1.width，k2 = frame2.height / frame1.height；

第二种方法是变化量转化法：对于有时效性的坐标系Oxy和Ox’y’，设某一时刻t，两个坐标系中坐标点分别为(x(t), y(t))，(x’(t), y’(t))；在t+1时刻，(x(t), y(t))变化为(x(t+1), y(t+1))，t ~ t+1时间内，Oxy坐标系中在x方向上的变化量为dx = x(t+1)-x(t)，dy = y(t+1)-y(t)，对于Oxy坐标系中的变化量，指定一个坐标值变化的范围，设该范围是一个矩形区域frame1，而假设Ox’y’坐标系中坐标值变化的矩形区域范围是frame2，那么可以将Oxy坐标系中坐标值变化量等比例映射到Ox’y’坐标系中去，即:

(‑2)

如此计算出在Ox’y’坐标系中x方向和y方向上的变化量，从而计算t+1时刻在Ox’y’坐标系中，由在坐标系Oxy上的点(x(t+1),y(t+1))转换而来的坐标值是(x’(t+1), y’(t+1))，其中x’(t+1) = x’(t) + dx，y’(t+1) = y’(t) + dy；

其中，第一种方法转换的特点是坐标点转换都是按等比例转换的，所以每一个时刻，两个坐标系中获取到的轨迹点是相似的，而且只要原坐标系中坐标点的数值没有发生突变，转换后的坐标点数值就不会发生突变，因而只要元坐标点足够精确，经过转换后的坐标点也是精确的；第二种方法中坐标点数值是基于坐标点变化量而言的，本身转换后的坐标点对于原坐标点的依赖性并不是太强，对于获取手势轨迹而言，往往会更加稳定，本文采用第二种方法对坐标点进行转换。

### 数据封装

从Kinect中获取的帧数据，是按照时序一帧一帧获取的，所以获取到的数据和时间是有关系的。在微软提供的Win32API函数中，调用GetTickCount这个函数可以获取系统时间，并且精确到毫秒。本文中，申明了一个结构体HandArgs，用来存当前时刻获取到手掌信息的各种参数：

typedef struct \_HandArgs

{

int x, y, z;

int vx,vy,vz;

int dx, dy,dz;

…

}HandArges

其中x,y,z的值是保存的当前时刻手掌心点的在各个方向上的坐标值，vx，xy，vz用来计算当前时刻和上一时刻掌心点在各个方向上的变化率，dx，dy，dz用来保存当前时刻和上一时刻掌心点在各个方向上的变化量。这些数据是通过以下步骤获取到的：

1. 定义两个时间参数currentTime和lastTime，分别记录当前时刻和上一时刻的毫秒数，并均初始化为0；
2. 从Kinect传感器中获取帧数据，对帧数据进行处理后，获取到手掌心点的坐标信息，将坐标点信息保存到HandArgs的x、y、z变量中并将当前系统时间的毫秒计数保存在currentTime中，。
3. 在判断lastTime非零的情况下，计算获取到两帧数据之间的时间差timeInterval = currentTime – lastTime，并获取HandArgs中保存的前一帧的x，y，z值，计算两者之间的差值，保存到dx，dy，dz，同时计算dx / timeInterval，dy / timeInterval，dz / timeInterval分别保存到vx，vy，vz中。

## 基于几何特征匹配的手势识别

不同的手势有不同的几何特征，包括手掌在某一特定时间内的坐标点的变化量和变化速率，双手之间的距离和双手关于y轴方向上形成的角度值等，根据这些几何特征可以利用数值算法对手势轨迹进行分类。

### 手掌的状态定义

在某一单独的特定时间、特定条件下，手势会呈现出不同的状态，本文中，定义了17种手势识别的状态：

enum GestureType

{

TYPE\_OPEN,

TYPE\_OPEN\_RETAIN,

TYPE\_OPEN\_RETAIN\_MOVE,

TYPE\_OPEN\_CLICK,

TYPE\_OPEN\_MOVE,

TYPE\_OPEN\_SHIFT,

TYPE\_HOLD,

TYPE\_HOLD\_MOVE,

TYPE\_UNKNOWN,

TYPE\_GRAB,

TYPE\_RELEASE,

TYPE\_DRAG,

TYPE\_ZOOM,

TYPE\_ZOOM\_IN,

TYPE\_ZOOM\_OUT,

TYPE\_ROTATE\_CW,

TYPE\_ROTATE\_CCW

};

这17中状态分别是：

1. TYPE\_OPEN：从Kinect2中能够获取到的手掌张开的状态；
2. TYPE\_OPEN\_RETAIN：手掌张开后，停留在屏幕上一段时间不动的状态；
3. TYPE\_OPEN\_RETAIN\_MOVE：手掌在屏幕上停留后在一段时间内移动的状态；
4. TYPE\_OPEN\_CLICK：手掌在屏幕上停留后在z方向上快速向前移动的状态，触发点击手势；
5. TYPE\_OPEN\_MOVE：手掌张开状态后移动的状态；
6. TYPE\_OPEN\_SHIFT：手掌张开后，在平面坐标系上快速移动的状态；
7. TYPE\_HOLD：手掌握拳状态，可以从Kinect2中读取到的状态；
8. TYPE\_HOLD\_MOVE：手掌握拳后在三维空间中移动的状态；
9. TYPE\_UNKNOWN：手掌未知状态；
10. TYPE\_GRAB：手掌从张开手掌到握拳变换的中间状态；
11. TYPE\_RELEASE：手掌从握拳到张开手掌变换的中间状态；
12. TYPE\_DRAG：手掌握拳后在平面坐标系上快速移动的状态；
13. TYPE\_ZOOM：手掌开始识别缩放旋转手势的状态；
14. TYPE\_ZOOM\_IN：双手放大手势状态
15. TYPE\_ZOOM\_OUT：双手缩小手势状态
16. TYPE\_ROTATE\_CW：双手顺时针旋转状态
17. TYPE\_ROTATE\_CCW：双手逆时针旋转状态

### 手势状态转换

有限状态机用于对系统的动态行为的建模，一般用状态图来可视化表示[1]。状态机是一个可以确定状态规则以及在特定条件下实现状态转移的设计模式，可以确定的是，手掌数据一定是处于一种既定的状态之下，在给定的外在条件下，会发生状态转移。在本文中，实现了手势状态转换的状态机，这样手势状态构成了一个有向图，如图 4‑1所示，手势状态通过条件的变化就会发生改变。在状态发生改变的一些时机就会触发手势事件。

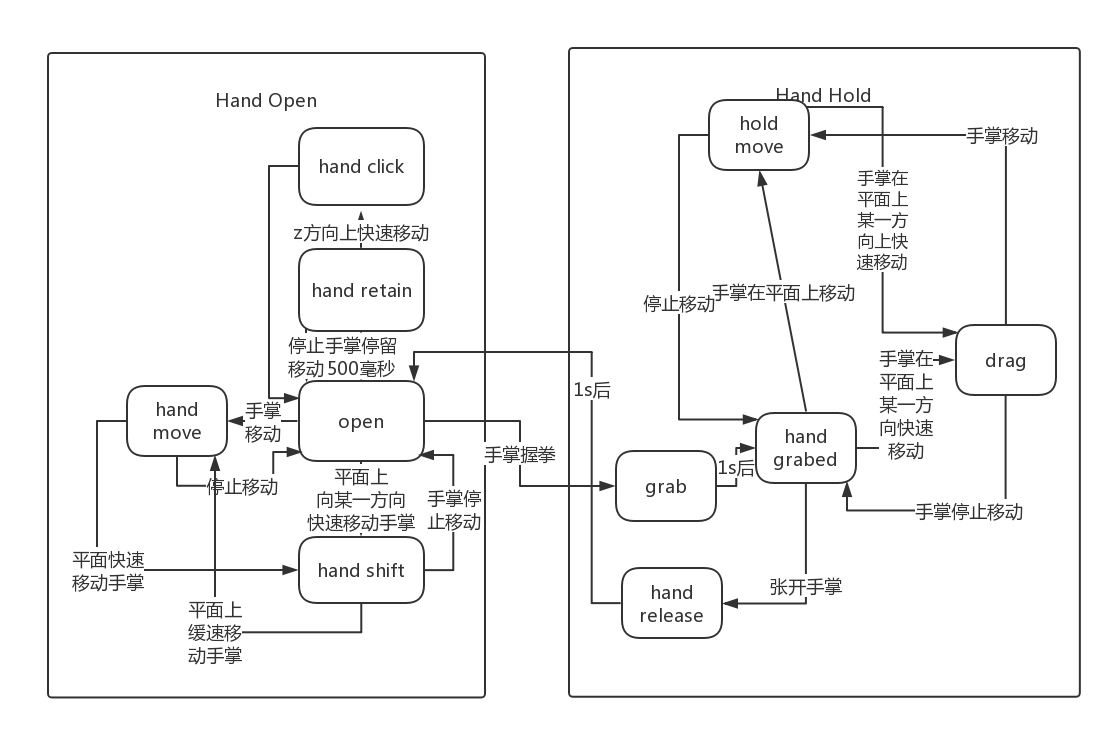


图 4‑1 手势状态转换图

### 几何特征匹配

对于单手的移动可以确定存在着三种类型：移动、快速移动、停留。根据获取到手掌参数，可以确定手掌在坐标系中的移动速度，移动变化量，再根据前后两帧之间的数值变化，可以对这三种单手移动类型进行判别。这几种手势移动的类型几何特征如下：

手掌移动：在通过Kinect的SDK中获取到手掌坐标点参数后，初始化一个

手掌快速移动：

手掌停留：

根据双手手掌轨迹数据的几何特征，可以确定双手手势的两种状态：双手缩放和双手旋转：同时

## DTW算法核心技术

一段具有时序的数据往往是指将同一统计指标的数值按其发生的时间先后顺序排列而成的数列，对于大部分的学科而言，时间序列是数据处理中一种十分常见的表达方式，比如一段视频数据可以看成是图片数据的时间序列、一段语音数据可以看成音节数据的时间序列等。在时间序列中，需要比较相似性的两段时间序列的长度可能并不相等，为了解决时间序列长度不同的模板匹配问题，DTW算法在上世纪70年代左右被提出来，并最早用于处理语音方面的识别分类问题。

DTW（Dynamic Time Wrapping，动态时间规整）算法是语音识别中的一种经典的算法[2]，该算法是基于动态规划（Dynamic programming，DP）的算法思想实现的。它能够简单灵活地实现模板匹配，能解决很多离散时间序列匹配问题。本文中获取到手势轨迹就是基于时间序列，和语音识别的过程是十分相似的，因此可以利用该算法实现对既定手势的匹配，达到比较精准地识别结果。

### DTW算法原理

假设有两个时间序列M和N，他们对应的长度（即帧数长度）分别为m和n，其中M是参考模板，它对应的序列设为{M1, M2, … ,Mi , … , Mm}，N是待匹配的模板，它对应的序列设为{N1, N2, … ,Nj, … , Nn}，规定一个距离函数Dis(M, N)，计算时间序列M和时间序列N之间的距离，由此可知，如果两个时间序列之间的距离越小，它们之间的相似度就越高。为了获取两个时间序列的距离，假设i和j分别表示M和N中任意选择的两个对应帧的帧号，则规定一个计算法则d，用d(Mi , Nj)表示这两帧之间的距离，在一般的DTW算法中，距离计算通常采用的距离是欧式距离。那么对于不同的m值和n值，有以下两种情况：

(1) 若m = n，说明序列M和序列N中帧数是一致的，这样可以直接计算两个序列一一对应的距离，即将M1与N1帧、M2和N2帧、… 、Mm和Nm帧对应起来，依次计算d(M1 , N1)，d(M2 , N2)，… ，d(Mm , Nn)的值，然后再求和，得到两个序列的距离值，如公式(4‑3)所示

(‑3)

(2) 若m≠n，说明序列M和序列N中的帧数是不一致的，对于这种情况，有两种可行的办法进行处理：

1. 采用线性缩放对齐的方法：将短的序列线性放大到和长的序列长度相同，然后再进行(1)中的距离计算，或者将长的序列线性缩短到和短序列一样的长度再进行(1)中的距离计算。

下面对此方法进行说明：假设m > n，也即模板序列M中的帧数长度大于待匹配序列N的帧数长度，这种情况下，可以将待匹配序列N用线性放大的方式，扩大成一个帧数长度为m的序列N’{N’(1), … , N’(n) }，然后再和M模板序列进行匹配，计算得到两者的距离。反之，m < n 的情况也可以通过这种方法获取到最终结果。

这种方法，虽然可以解决不同长度的时间序列之间的匹配问题，但是对于不同的时间序列，在各段的不同情况下会产生或长或短的变化，即各个序列中元素的伸缩程度是不同的，如果采用这种方法，就忽略了这种不同伸缩程度带来的变化，而统一地使用线性收缩的方式使得两个时间序列的长度变得相同，势必会造成识别的结果精度不准，为了更佳的识别效果，一般会采用动态规划的方法；

2．采用动态规划的方法：为了将这两个序列对齐，动态规划算法会构造一个m×n的矩阵网格，矩阵中元素(i, j)表示序列M中第i帧和序列N中第j帧的距离，用d(Mi, Nj)来表示，这个距离越小，表示这两个序列中对应的这两帧之间的相似度越高，也可以理解为失真度越低。

由于这两个序列是有时间顺序的，也就是说整个匹配的过程，必定是要寻找一条路径，经过矩阵网格的(1,1)到达矩阵网格的(m,n)。直观来看，如果把序列M的帧号1~m放到一个二维坐标系的横轴，做为横轴的标号，把N的帧号1~n放到这个坐标系的纵轴，作为纵轴的标号，那么动态规划的方法归结为从这个坐标系中找到一条路径，使得这条路径能够从从左下角到达右上角，并且这条路径经过的元素值之和最小。

为了方便描述这条路径，假设这条路径经过的点依次是(X1,Y1)，… ，(Xi,Yj)，… (Xm, Yn)，其中(X1,Y1) = (1,1)，(Xn, Yn) = (m, n)，定义一个二维函数φ(i, j)表示到达点(Xi, Yj)时，与邻近点元素积累的距离之和的最小值，那么满足公式(4‑4)

(‑4)

这个公式体现了局部最优路径的选择，那么从匹配起点就能沿着局部最优的路径反向找到最初的匹配点，获的全局最优路径。这种逐点求取帧匹配距离的算法就是DTW算法。

### DTW算法的约束条件

根据上面介绍的DTW算法基本原理，对于满足可以用DTW算法匹配的两段时间序列M(M1, M2, … , Mi, ... , Mm)和N(N1, N2, … , Nj, ... , Nn)，设最后找到的满足两者之间距离和最小的路径L，这条路径一共经过s个点且Li表示这条路径上面的第i个点（1≤i≤s），那么它将满足三个约束条件：

1. 边界条件唯一性：这条路径的起点必定是(1, 1)，终点必定是(m, n)，即L1=(1,1)，L2=(m, n)；
2. 连续性：对于路径L，如果第k（1≤k＜s）个点Lk为(i, j)，那么在路径L上第k+1个点必定是(i+1, j)、(i, j+1)、(i+1,j+1)之中的一个；
3. 单调性：根据条件(2)，路径L上第k（1≤k＜s）个点(ik, jk)和第k+1个点(ik+1,jk+1)，必定满足：ik≤ik+1且jk≤jk+1

这三个约束条件保证了在上面构造的m×n的矩阵网格中每一个点的路径只有三个方向，如图 4‑2所示

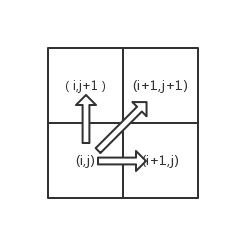


图 4‑2 DTW矩阵网格最优路径的三个方向

也就是说，若最优路径通过了网格中的点(i, j)，那么在最优路径的下一个网格点必定只有三种情况：(i, j+1),(i+1, j),(i+1, j+1)

### DTW算法实例

下面用一个实例来说明DTW算法的计算过程，假设有模板时间序列M{3,5,2,9,10}，有两个待匹配的时间序列N1{4, 13,12,12,6,4}，N2{1,6,3,12}，它们的帧长度分别为m=5、n1=6、n2=4，它们的时序图如图 4‑3所示，

|  |  |  |
| --- | --- | --- |
|  |  |  |
| (a) M序列时序图 | (b) N1序列时序图 | (c) N2序列时序图 |

图 4‑3 样例时序图

那么计算M序列和N1序列的距离值（相似度）的DTW算法过程如下：

第一步、构造距离矩阵Dis[m][n1]，为方便理解，距离矩阵的右方向是i值的递增方向，上方向是j值的递增方向，Dis[i][j]中计算M中第i帧和N1中第j帧的欧式距离，即两个帧数据值之间的差值的绝对值，那么构造的距离矩阵如图 4‑4所示

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 1 | 2 | 5 | 6 |
| 3 | 1 | 4 | 3 | 4 |
| 9 | 7 | 10 | 3 | 2 |
| 9 | 7 | 10 | 3 | 2 |
| 10 | 8 | 11 | 4 | 3 |
| 1 | 1 | 2 | 5 | 6 |

图 4‑4 序列M和序列N1的距离矩阵

第二步、构造累加求和距离矩阵D[m][n1]，D[i][j]表示经过(i, j)是累加的最小距离值，根据公式(4‑4)，计算所得的结果如图 4‑5所示

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 33 | 25 | 26 | 22 | 22 |
| 32 | 24 | 27 | 17 | 16 |
| 29 | 23 | 26 | 14 | 12 |
| 20 | 16 | 19 | 11 | 10 |
| 11 | 9 | 13 | 8 | 11 |
| 1 | 2 | 4 | 9 | 15 |

图 4‑5 序列M和序列N1的累加求和距离矩阵

右上角的数值D[m][n1]=22就是最后所求的序列M和序列N1的之间的距离，根据箭头指示的方向，由DTW算法求得的这条路径是(1,1)→(2,1)→(3,1)→(4,2)→(5,3)→(5,4)→(5,5)→(5,6)。

计算M序列和N2序列的距离值过程和上述算法相同，第一步构造的距离矩阵如图 4‑6所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 9 | 7 | 10 | 3 | 2 |
| 0 | 2 | 1 | 6 | 7 |
| 3 | 1 | 4 | 3 | 4 |
| 2 | 4 | 1 | 8 | 9 |

图 4‑6 序列M和序列N2的距离矩阵

第二步中构造的累加求和距离矩阵如图 4‑7所示，求得两个序列之间的距离为D[m][n2]=9,再根据箭头的方向，算得距离最短的路径是(1,1)→(2,2)→(3,3)→(4,4)→(5,4)。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 14 | 12 | 14 | 7 | 9 |
| 5 | 5 | 4 | 10 | 17 |
| 5 | 3 | 7 | 10 | 14 |
| 2 | 6 | 7 | 15 | 24 |

图 4‑7 序列M和序列N2的累加求和距离矩阵

由于Dis{M，N1} > Dis{M，N2}，说明N2与M的距离比N1与M的距离更小，即N2与M更相似。

## 基于DTW算法的手势识别

前文提到，手势轨迹是一个时间序列，它必定是一个有起点和终点，一段时间内有序的手掌坐标点数据的集合。因此，借鉴语音识别的案例，DTW算法也可以用来进行手势识别。

由于DTW算法是一种基于模板匹配的算法，所以需要提供样本模板数据，建立模板手势库。在已规定好的轨迹特征模板手势库的情况，再用DTW算法对手势和手势库中的模板数据进行匹配，才能有效地识别出特定的手势。

### 建立模板手势库

对于手势库的建立，首先第一点就是要设计几种已经可以实际使用的手势，并把手势的特征轨迹点记录下来。考虑到易于理解和易于扩展手势库的使用场景，手势库数据使用xml文件来记录，xml的层次结构图 4‑8所示，即xml的根节点手势库（gestureLib）保存了多个手势(gesture)，手势又是由多条轨迹（stroke）组成，一个轨迹是有多个轨迹坐标点（point）组成，如此就构建了一个保存样本模板的xml数据库文件。

图 4‑8 手势库xml文件的层次结构图

构建xml文件的一个显著的优点就是文件易于理解，易于编辑，可读性高。但是对于大量数据的解析会比较缓慢，且文件格式复杂，如果读取到内存中会占据大量不必要的格式标签信息，因此对于读取到内存中的模板文件信息，定义了一个协议规范，将xml文件转换成二进制的model文件，用以缩减内存存储消耗。

Model文件的协议规范：首先定义了一个ModelHeader，用来保存模板文件的基本信息，结构体定义如下：

struct ModelHeader

{

char sign[4];

short ver;

short type;

short width;

short height;

short pointDist;

int count;

};

其中，各个变量的意义是：

sign：模板文件的标志，是一个字符串，用来标识模板文件，默认为“SGR”，是简单手势识别（simple gesture recognizer）的意思；

ver：模板文件的版本号，用来标识模板文件的版本；

type：模板文件的类型；

width：手势轨迹的默认宽度；

height：手势轨迹的默认高度；

pointDist：手势轨迹之间默认的距离差值，用以方便把轨迹稀疏化或者补全离散轨迹的临界阈值；

count：模板文件中已经定义了的手势个数。

然后对于各个手势，定义了一个UTF-8字符来标识手势轨迹的信息，称为手势标识码，比如，用○（UTF-8编码为&#x25CB;）来标识圆圈手势。在程序设计中定义FeatureHeader和Feature结构体与各个手势对应，FeatureHeader结构体中存储了各个手势的基本信息，而Feature结构体中定义了手势的内容，包括手势坐标点的信息。他们的结构体定义如下：

struct FeatureHeader

{

char word[4];

int trackCount;

int count;

};

struct Feature

{

int trackCount;

string word;

vector<int> feature;

};

其中word表示手势轨迹的标识码，trackCount表示该单一手势的轨迹数目，设置这个轨迹数目

### 模板手势库轨迹处理

## 手势识别的模块设计

## 本章小结

本章主要从

# 系统框架搭建和测试

## 软件系统的实现框架

软件开发环境

软件设计流程图

具体的实现过程

## 几种手势的应用场景

## 手势识别过程分析

## 本章小结

# 总结与展望

## 总结

## 展望和改进

致谢

**校对报告**

当前使用的样式是 [硕士论文]

当前文档包含的题录共2条

有0条题录存在必填字段内容缺失的问题

所有题录的数据正常

**References:**

[1] 徐小良, 汪乐宇, 周泓. 有限状态机的一种实现框架. 工程设计学报, 2003, 05: 251-255

[2] 童红. 孤立词语音识别系统的技术研究:[ 硕士学位论文]. 江苏大学, 2009.