

# 上海大学虚拟校园漫游系统



上海大学  
通信与信息工程学院

SHU / School of Communication & Information Engineering

学 生 1: 苟凯俞 11122933

学 生 2: 杜旦阳 11122949

学 生 3: 黄玮奇 11123584

组 长: 苟凯俞 11122933

组长电话: 18221050966

指导老师: 高永章, 王瑞

完成时间: 2014 年 10 月

# 目录

摘要.....	3
一、 课题概述.....	4
二、 课题实施方案.....	4
2.1 方案说明 .....	4
2.2 工程规范 .....	6
三、 课题实施过程详述.....	9
四、 系统测试方法.....	17
五、 结果分析.....	20
六、 总结.....	21

## 摘要

本课题以上海大学宝山校区为依托，基于时下用户群体最广阔的 android 开发平台，建立移动终端上使用的虚拟校园漫游系统。该系统具有支持 360 度全景场景展示，基于姿态的体感控制，热点跳转等创新性功能。可以方便查看校园环境，达到虚拟与现实的交互。PC 端采用 MATLAB 编程进一步解决全景图片合成的算法问题。整个研发过程，符合相关软件编程规范。最终开发出了一款功能强大，效果突出的全景浏览程序。本项目为解决原始图片浏览的视角狭窄问题提供了解决方案，方便了用户对校园信息的访问，从而可以促进大学的建设和远程教学的发展。

**关键词：**全景浏览，图像拼接，虚拟之旅

## Abstract

The topics to Shanghai University Baoshan campus-based, user groups based on today's most extensive android development platform to create a virtual campus roaming system for use on mobile devices. The system has the support of 360-degree panoramic scenes show, based on a sense of control, and other innovative features hot jump posture. You can easily view the campus environment, to interact with the virtual and reality. PC side using MATLAB programming algorithm to solve the problem further panoramic picture synthesis. Throughout the development process, in line with the relevant software programming specifications. Eventually developed a powerful effect prominent panoramic viewer. This project is to solve the original problem of a narrow angle image viewer provides a solution to facilitate the user's access to campus information, which can promote the development of university building and distance learning.

**Keywords:** Panoramic view, image stitching, virtual tours

## 一、课题概述

本课题以上海大学宝山校区为依托，建立基于 Android 平台的虚拟校园漫游系统及图像合成算法的研究。以下是本项目实施的基本功能与核心问题。

### 1. 图像处理功能：

1.1 通过基于 MATLAB 的图像处理算法，对得到的图像进行全景合成。

1.2. 通过优化算法对素材图像进行优化处理，解决图像合成中的边界模糊问题。

### 2. 移动终端的展示功能：

2.1. 支持球形，立方和柱面全景图像

2.2 允许滚动和连续滚动

2.3. 支持从左向右滚动，并从顶部到底部使用加速度计

2.4. 允许使用惯性停止滚动

2.5. 支持放大和缩小（移动两个手指在屏幕上）

2.6. 支持复位（放置三根手指在屏幕上或摇动设备）

2.7. 允许您控制在 x 和 y 轴旋转摄像头的范围

2.8. 支持热点跳转

2.9. 感官的旋转 - 支持（仅兼容，加速度计和磁力计的设备），比如您不需要按键，只需要摇晃手机就可以看到不同方向的全景图。

本课题的核心问题在于 1. 针对获得的平面图像进行拼接。2. 基于 android 手机平台上全景应用程序的编写以及对上述功能的实现。主要难点在于图像拼接算法和面向对象的 JAVA 编程。

## 二、课题实施方案

### 1.1 图像拼接方案.

图像的匹配及拼接是全景图像生成技术的关键。考虑到 SIFT 特征匹配算法具有很强的匹配能力，可以处理两幅图像之间发生平移、旋转、仿射变换情况下的匹配问题，所以课题组决定用以 SIFT 算法为基础的图像拼接算法实现图像的合成处理。

SIFT 算法的实质是在不同的尺度空间上查找特征点，并计算出关键点方向。它所查找到的关键点是那些不会因光照、仿射变换和噪音等因素而变化的点，如角点、边缘点、暗区的亮点及亮区的暗点等。在特征点向量匹配后，调用白平衡、combine 等模块函数对图像进行再处理，最终即可实现图像的拼接合成。

### 1.2 终端展示方案.

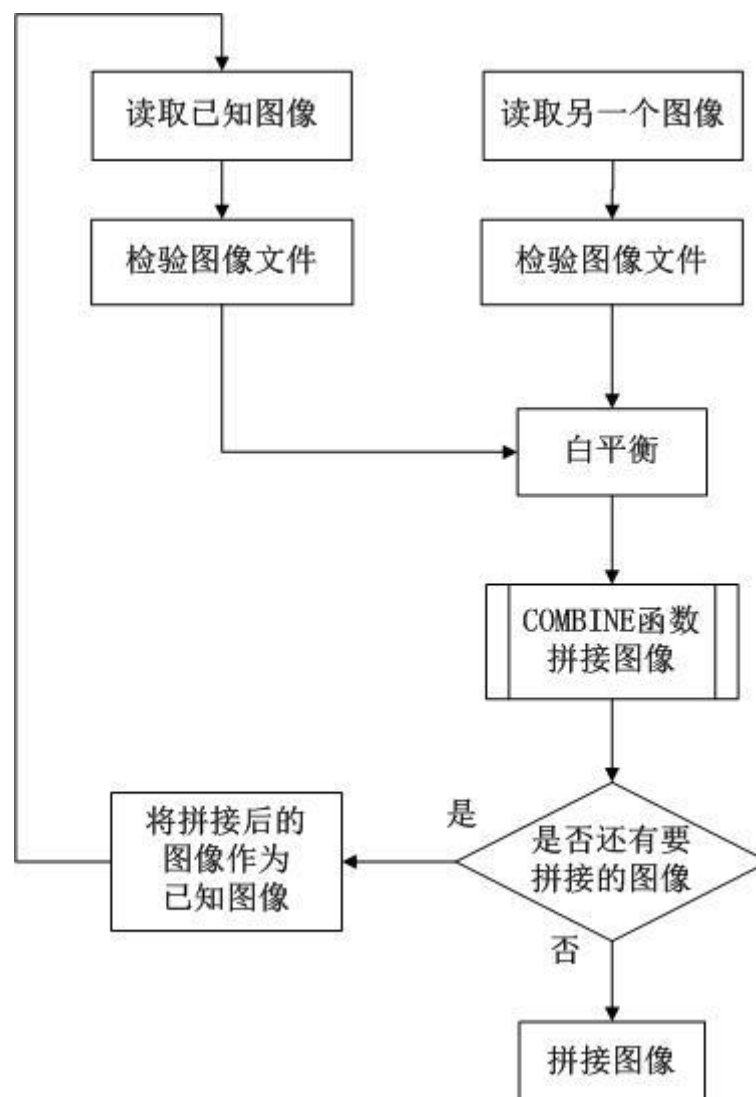
考虑到手机应用的时新性，流传性和便捷性，课题组决定展示终端订为手机平台。操作系统的选择方面，考虑到 android 平台庞大的用户群体，编程的自由度较高以及整体程序架构的稳定性和可移植性更好，决定选择基于面向对象的 JAVA 语言开发全景展示程序。

在 Android 平台上进行应用开发,设计实现校园虚拟现实。加入信息提示和互动功能。通过 JAVA 语言建立不同的类别,对应不同的应用功能层面,有系统自己的 library 资源库,这样做创新在于把项目分层处理,从底端到高端的开发,更能满足消费者的需求,而且后期在进行周边应用开发的时候,我们有很多可以应用的库与接口,使得系统高度集成,优化方便。

## 2.1 方案说明

### 2.1.1 图像拼接算法

本项目的第一部分工作是在 matlab 平台上进行的,目标是将 2 张及以上,大小一样的照片拼接在一起。作为下一部分应用于 android 平台的材料。实现图像拼接部分的程序整体设计思路如下:



- (1) 待处理的图像在程序运行之前,存放在 **image** 文件夹内,方便主函数的调用,图像的像素大小必须一致,名字按照顺序分别命名为 1、2、3、4、5、6,以 JPG 格式保存。

- (2) 所有的起作用的子函数整合在 run.m 文件中，点击 run.m 即可运行，可以自动得到几张图像拼接的最终结果，并且可以在 command 窗口输入相应的函数，查看中间的处理过程。
- (3) 处理完成后只需将最终结果导出，即可作为成品应用到 android 平台上。

## 2.1.2. 全景展示方案

移动终端开发基于 Eclipse 开发软件,运用面向对象的 JAVA 语言,采用自顶向下的项目编程思想,全部解决了目标的九项功能,整体设计思路与软件框架见图 1。

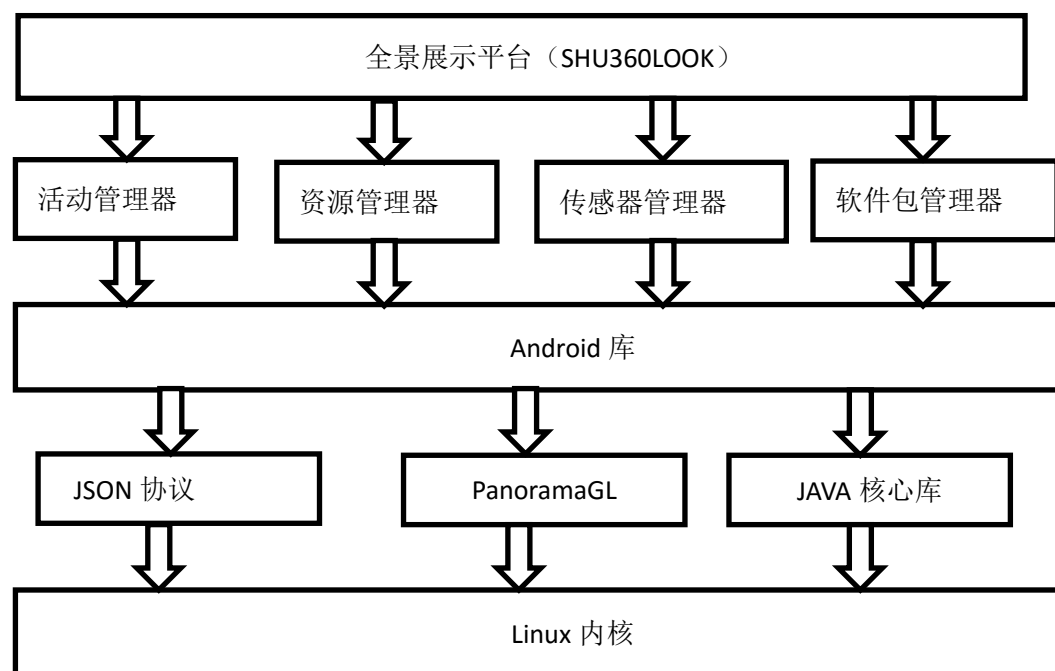


图 1. 全景展示平台开发架构

从该框架可以看出由用户决定的功能到具体的软件思想实现的整个过程。包括了：1. 由功能决定选用了 PanoramaGL 开发库（全景解决方案）；2. 由应用程序的可操作性和具体功能出发，编写基于 PanoramaGL 库下的终端程序；3. 考虑到软件的可移植性和可扩充性，项目组决定使用 JSON 协议。4. 用户节目的设计以及可操作性，均使用 android 自带库进行开发。

在后期测试阶段，考虑到 android 平台下的手机类型众多，我们提供了从 android4.0 版本到 andorid4.3 版本的解决方案，以使得更多的用户能够创建自己的全景之旅。

## 2.2 工程文件介绍和代码规范

### 2.2.1. MATLAB 部分：

#### (1) Matlab 部分总体介绍

核心程序由三部分组成，整个部分就是不断调用这个三部分的过程：

1. 图像读入，存储和白平衡处理
2. Sift 算法在 matlab 平台上的实现，能够提取出图像的特征描述
3. 利用两张图像之间对应的特征描述点完成拼接过程

## (2) 关键函数介绍

1. run.m 作为总文件，是整个工程最顶层的文件和应用的入口
2. imread.m 读取图像
3. imresize.m 图像缩放
4. whiteBalance.m 对图像进行白平衡处理
5. Combine.m 对整个sift文件进行整合，实现图像拼接
6. sift.m 提取图像的特征描述点
7. siftmatch.m 根据两张图像的特征描述点实现图像的匹配

**注：**这些关键的函数在总体上实现整部分工程，其余的代码都是作为子函数，在以上这个几个关键的函数文件中被调用

## 2.2.2. JAVA 及 JSON 协议部分

### 一、android 开发工程规范

一个软件的生命周期中，80%的花费在于维护。编码规范可以改善软件的可读性，可以让程序员在二次开发时尽快而彻底地理解新的代码。本项目为了实现产品的市场化，在编程中均按照如下规范操作：

### (一) 命名规则

1. 一个唯一包名的前缀总是全部小写的 ASCII 字母并且是一个顶级域名，通常是 com, edu, gov, mil, net, org。
2. 类名是个名词，采用大小写混合的方式，每个单词的首字母大写。接口一般要使用 able、ible、er 等后缀。
3. 方法名是一个动词，采用大小写混合的方式，第一个单词的首字母小写，其后单词的首字母大写。例如：`public void run(); public String getBookName();`
4. 变量命名，第一个单词的首字母小写，其后单词的首字母大写。变量名不应以下划线或美元符号开头，尽管这在语法上是允许的。变量名应简短且富于描述。变量名的选用应该易于记忆。
5. layout 命名, layout xml 的命名必须以全部单词小写，单词间以下划线分割，并且使用名词或名词词组，即使用 view 缩写\_模块名称\_view 的逻辑名称来命名。
6. id 命名, layout 中所使用的 id 必须以全部单词小写，单词间以下划线分割，并且使用名词或名词词组，并且要求能够通过 id 直接理解当前组件要实现的功能。

### (二) 代码风格。

1. 缩进规约：不允许使用 Tab 进行缩进，使用空格进行缩进，推荐缩进为 2 空格。

2. 空行, 空行将逻辑相关的代码段分隔开, 以提高可读性。下列情况应该总是使用空行: 一个源文件的两个片段(section)之间; 类声明和接口声明之间; 两个方法之间; 方法内的局部变量和方法的第一条语句之间; 一个方法内的两个逻辑段之间, 用以提高可读性。
3. 行宽: 无特别规定, 因为现在的显示器都比较大, 所以推荐使用 120 进行设置。
4. 方法: 一个方法尽量不要超过 15 行, 如果方法太长, 说明当前方法业务逻辑已经非常复杂, 那么就需要进行方法拆分, 保证每个方法只作一件事。不要使用 try catch 处理业务逻辑。
5. 参数和返回值一个方法的参数尽可能的不要超过 4 个。如果一个方法返回的是一个错误码, 请使用异常。尽可能不要使用 null, 替代为异常 或者使用空变量 如返回 List 则可以使用 Collections.emptyList()。

### (三) 具体代码规范

1. onCreate 中完成控件的初始化 ✓ onResume 中完成数据的准备。
2. 重复逻辑需要抽象成方法 ✓ 放在类的代码中调用 □ 多个类都需要使用到的方法可以做成工具类 ✓ 放在 \*.utils 里 □ 大量代码的方法重写 ✓ 放在 \*.adapter 里。
3. 应尽量避免使用类变量。一般来说, 变量的作用域越小越好。为了减少类变量, 方法之一是将数据作为参数在不同方法之间传递, 而不是让方法共享类变量。
4. java 代码中不要出现中文。
5. 程序中应该有足够的调试信息, 一般来讲调试信息必须指明此信息是在哪个文件、哪个方法、哪一行输出的。调试信息的文字描述不应采用缩写的方式, 应当尽可能的描述清楚此调试信息的目的。
6. styles.xml: 将 layout 中不断重现的 style 提炼出通用的 style 通用组件, 放到 styles.xml 中。
7. 如果多个 Activity 中包含共同的 UI 处理, 那么可以提炼一个 CommonActivity, 把通用部分叫由它来处理, 其他 activity 只要继承它即可。
8. 当快速滑动列表时 (SCROLL\_STATE\_FLING), item 中的图片或获取需要消耗资源的 view, 可以不显示出来; 而处于其他两种状态 (SCROLL\_STATE\_IDLE 和 SCROLL\_STATE\_TOUCH\_SCROLL), 则将那些 view 显示出来。
9. 数字、字母和汉字混排占位问题: 将数字和字母全角化。由于现在大多数情况下我们的输入都是半角, 所以字母和数字的占位无法确定, 但是一旦全角化之后, 数字、字母的占位就和一个汉字的占位相同了, 这样就可以避免由于占位导致的排版问题。

## 二、JSON 协议开发规范

1. 和 XML 一样, JSON 也是基于纯文本的数据格式。由于 JSON 天生是为 JavaScript 准备的, 因此, JSON 的数据格式非常简单, 您可以用 JSON 传输一个简单的 String, Number, Boolean, 也可以传输一个



数组，或者一个复杂的 Object 对象。String, Number 和 Boolean 用 JSON 表示非常简单。例如，用 JSON 表示一个简单的 String “abc”，其格式为：“abc”除了字符 “\”, “/”, “/” 和一些控制符 (\b, \f, \n, \r, \t) 需要编码外，其他 Unicode 字符可以直接输出。

2. Boolean 类型表示为 true 或 false。此外，JavaScript 中的 null 被表示为 null, 注意, etrue、false 和 null 都没有双引号, 否则将被视为一个 String。
3. Object 对象在 JSON 中是用 {} 包含一系列无序的 Key-Value 键值对表示的，实际上此处的 Object 相当于 Java 中的 Map<String, Object>, 而不是 Java 的 Class。注意 Key 只能用 String 表示。
4. 简单示例。

一个 Person 对象包含 name 和 address 对象，可以表示如下：

```
{"name": "Michael", "address":  
    {"city": "Beijing", "street": "Chaoyang Road",  
    "postcode": 100025}  
}
```

## 三、课题实施过程详述

### 3.1 图像拼接算法

#### 3.1.1 基于本项目 MATLAB 的优点阐述

- 1) 代码接近 C 语言，方便学习和使用；
- 2) 互联网上有丰富的开源代码资源，可以供我们调用；
- 3) 用现成的代码作为子函数，可以随时在 command 界面调用，以及查看变量，方便修改和测试；
- 4) 模块化，适合结构分块清晰的工程；
- 5) 支持 MATLAB 与 C 语言的混编，C 文件运行速度快，而 MATLAB 语言编译的文件方便调用；

#### 3.1.2 方案原理

本项目运用的是以 SIFT 算法为基础的图像拼接算法。首先利用 SIFT 算法提取图像的特征点并进行匹配，然后通过编写 combine、whiteBalance、siftmatch 等模块函数实现图像的拼接。

这里给出 run.m 的代码：

```
clc  
clear all  
addpath('sift');  
addpath('filters');  
  
img_1 = imread('images/1.jpg');
```

```
img_2 = imread('images/2.jpg');
img_3 = imread('images/3.jpg');
img_4 = imread('images/4.jpg');
img_5 = imread('images/5.jpg');
img_6 = imread('images/6.jpg');

% testing
img_1 = imresize(img_1, 0.3);
img_2 = imresize(img_2, 0.3);
img_3 = imresize(img_3, 0.3);
img_4 = imresize(img_4, 0.3);
img_5 = imresize(img_5, 0.3);
img_6 = imresize(img_6, 0.3);

% bilanciamento del bianco
img_1 = whiteBalance(img_1);
img_2 = whiteBalance(img_2);
img_3 = whiteBalance(img_3);
img_4 = whiteBalance(img_4);
img_5 = whiteBalance(img_5);
img_6 = whiteBalance(img_6);

figure(), imshow(img_1);
figure(), imshow(img_2);
figure(), imshow(img_3);
figure(), imshow(img_4);
figure(), imshow(img_5);
figure(), imshow(img_6);

c = combine(img_1, img_2);
c1 = combine(c, img_3);
c2 = combine(c1, img_4);
c3 = combine(c2, img_5);
c4 = combine(c3, img_6);

unsharpFilter = fspecial('unsharp');
c4 = imfilter(c4, unsharpFilter);
figure(), imshow(c4);
```

### 3.1.3 SIFT 算法步骤

SIFT 算法流程如图 2:

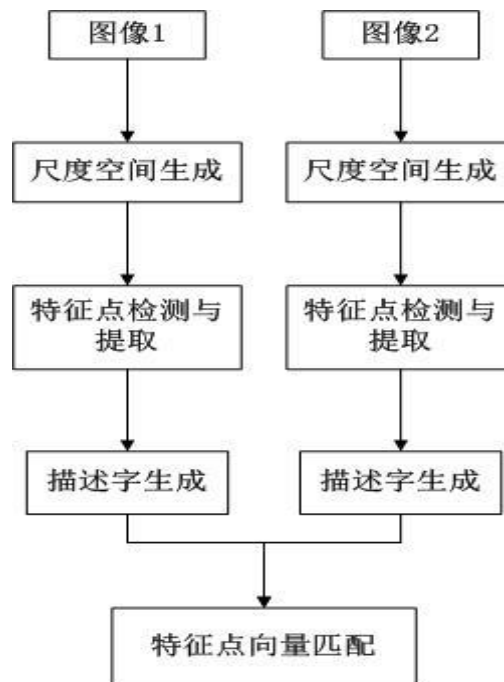
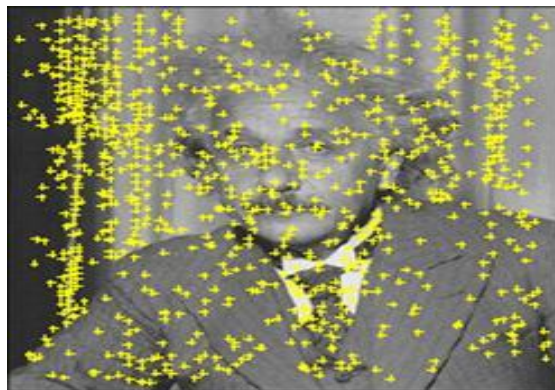


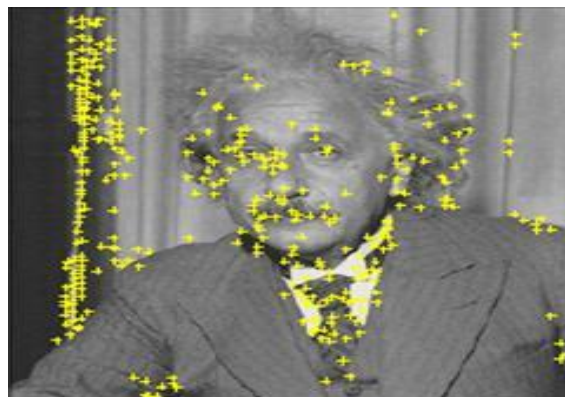
图 2. SIFT 算法流程图

关于上述进程，如下是详细示例。

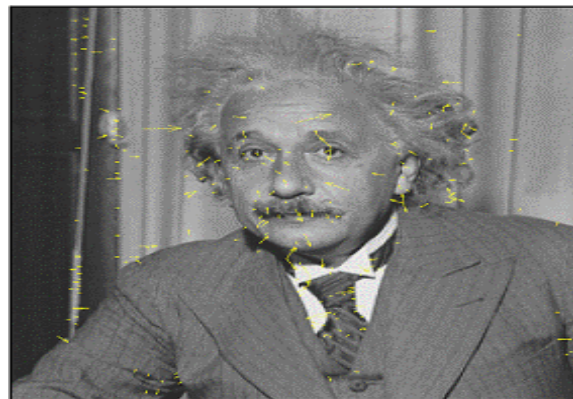
1) 构建尺度空间，检测极值点，获得尺度不变性；



2) 特征点过滤并进行精确定位；

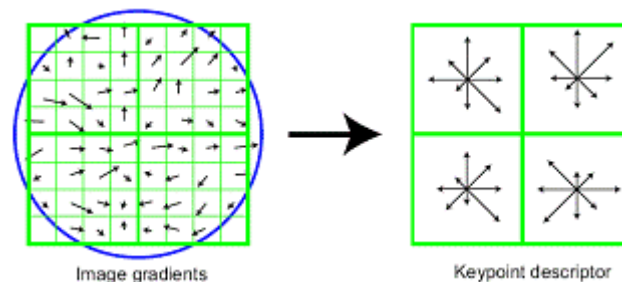


3) 为特征点分配方向值;



4) 生成特征描述子。

以特征点为中心取  $16 \times 16$  的邻域作为采样窗口, 将采样点与特征点的相对方向通过高斯加权后归入包含 8 个 bin 的方向直方图, 最后获得  $4 \times 4 \times 8$  的 128 维特征描述子。示意图如下:



当两幅图像的 Sift 特征向量生成以后, 下一步就可以采用关键点特征向量的欧式距离来作为两幅图像中关键点的相似性判定度量。取图 1 的某个关键点, 通过遍历找到图像 2 中的距离最近的两个关键点。在这两个关键点中, 如果次近距离除以最近距离小于某个阈值, 则判定为一对匹配点。

下面的对比图是 Sift 特征匹配的例子:



5) 特征提取函数的使用

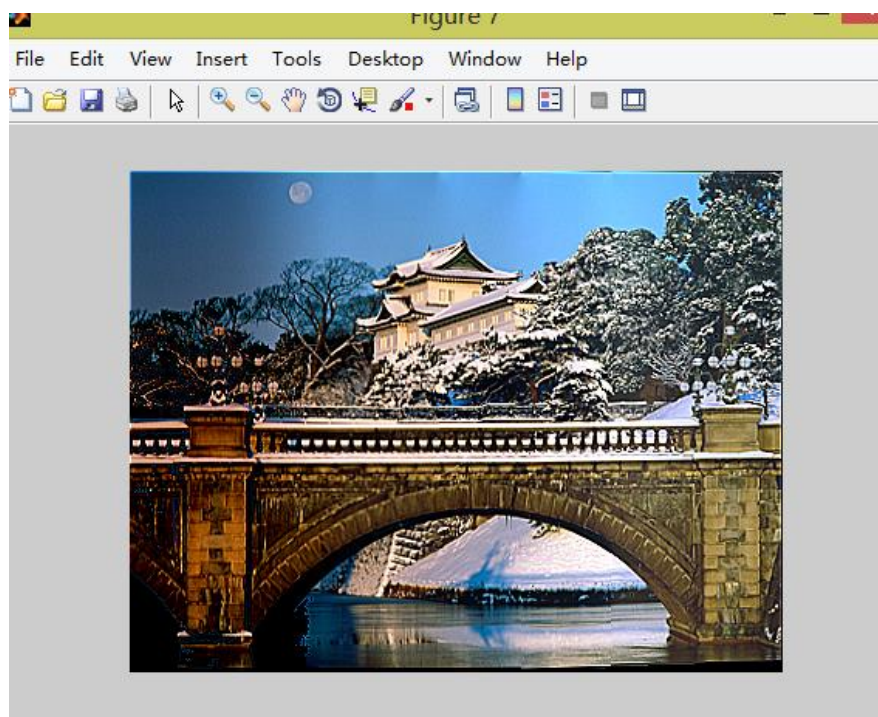
作为一种匹配能力较强的局部描述算子, SIFT 算法的实现相当复杂, 但从软件开发的角度来说, 只要会使用其中几个比较重要的函数就行了。



#### 3.1.4 图像拼接算法

在特征点向量匹配后，调用白平衡、combine 等模块函数对图像进行再处理，最终即可实现图像的拼接。

以六张图像的拼接做测试，可得到以下合成结果：



3.2 全景展示:

3.2.1 主函数设计详细介绍.

主函数是是面向对象程序开发的最核心部分，程序运行的主要构架均体现在主函数的进程上。

主函数运行进程如图 3 所示：

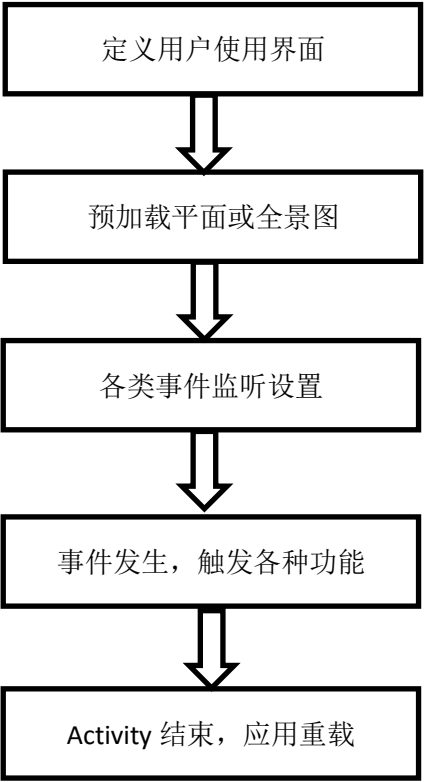


图 3. Android 程序主函数进程

关于上述进程，如下是详细解释。

第一．人机交互程序要想取得很好的用户反馈，操作界面必须简介，方便。因此小组决定采用直进入式，列表选择，线性铅直排列的布局。从而定义了的功能控件包括下拉列表（用到下拉容器），放缩控件（采用加减式）和热点跳转触发键（虚拟）。

第二．本程序的主要亮点在于能够提供给用户前所未有的全景体验。基于此设计思想，我们提供了平面，球体 1（1024x512），球体 2（2048x1024）以及柱体共 4 种浏览方式。鉴于程序的大小以及推广的可携带性，目前决定把最具有代表性的几种全景图放入我们的程序中。

第三. 通过以上两步，我们得到了单个的全景浏览，但要成为虚拟全景之旅，经过讨论，我们摒弃了列表式的跳转方式。采用单击箭头视角跳转的方式，通过给不同的场景进行编号，加入相应的场景过渡效果，使用户更能体验到身临其境的感觉。

第四，关于全景图像的加载，热点跳转，我们均采用了主函数调用相应库函数的接口实现。主要包括：panoramagl.loaders, panoramagl.spots 等封装库。

第五，考虑到现今手机操作系统的多样性，项目组决定使用 JSON 协议编写程序四组全景方案的主要功能。因为 JSON 是一种轻量级的数据交换协议。不仅

仅是 android 系统，包括 ios 系统在内的终端上也可以运行基于 JSON 协议的应用程序。另外使用基于 JSON 的编程可以使得程序的可扩展性大幅度提高，后续开发者可以在我们的基础上进行功能上的进一步开发。

### 3.2.2JSON 协议详解

本程序 JSON 协议的父类是 json\_cubic.data, 其他协议均是在此基础上继承了父类的相关功能，添加了不同的素材。因此，这里主要分析该文档。首先定义了 JSON 的数据交换格式，然后运用 android 自带库以及 panoramagl 库对协议进行解析，以实现数据交换。

在立方体模式的 JSON 中，我们定义了如下功能：

1. "urlBase": "res://raw",，即加载图像的来源，目前支持从手机 SD 卡读取图像以及从服务器的云端下载图像。考虑普通用户不一定需要网络连接，这涉及到隐私保护问题，故本程序采用直接从程序的 raw 文件中读取素材。

2. 定义了三种全景浏览类型，分别是立方体，球体以及柱体。这里以立方体为例进行说明。调用 PLCubicPanorama.java 库的接口实现该功能，主要进程为：定义相关类与变量→获得拼接成 cubic 形式所需要的图片数（6）→通过 setTexture 把获得的六张图片拼接，此过程中程序自动加入图片拼接，不断增大图片长和宽，直到最终拼接完成。其他两类全景模式与上述过程相似，区别在于加载的图像是成型的全景图，并不需要程序做拼接工作。

3. 定义了所预览图像的分辨率。通过 android 自带库可以方便解析，本程序包括的分辨率有：1024x512，2048 x1024，1024 x1024，512 x256。

4. 定义了复位功能，包括摇动复位（具有摇门槛），以及三指触碰复位。

5. 设置手指滑动操作功能，用滑动灵敏度来衡量。通过实时监听机制，监听使用者是否触屏，相关方法继承 PLJSONLoader.java 中有关 scrolling 的判断语句。

6. 设计基于重力的加速度控制功能，使图片具有上下移动以及左右移动的功能。

7. 明确立方体全景图像模式（上下左右前后六张图片分别对应的图像序号）。

8. 设置摄像头，即可以通过设置使得用户使用时所看到的视场角度范围可变，本程序采用左右方向视角为 360 度，上下方向视角为 180 度。

9. 定义了放缩功能的缩放级别与缩放系数，可以实现更好的浏览与人机交互。

10. 定义热点数组，主要是为了方便热点跳转时候，图片能有顺序，按设计者的设置进行跳转，方便程序推广。

以下为本程序 JSON 协议典型代码：

```
{
    "urlBase": "res://raw",    // URL基地所在的文件
    "type": "cubic",          //立方体全景
    "keep": "all",            //使能所有功能
    "imageColorFormat": "RGBA8888",
    "reset":                  //复位
    {
        "enabled": true,
        "numberOfTouches": 3,    //复位所需触摸点
        "shake":
```

```

{
    "enabled": true,
    "threshold": 1300    //摇门槛复位
}
},
"scrolling":           //滚动操作
{
    "enabled": true,
    "minDistanceToEnableScrolling": 30
},
"inertia":             //惯性节
{
    "enabled": true,
    "interval": 3.0
},
"accelerometer":      //加速度计
{
    "enabled": false,
    "interval": 0.033,
    "sensitivity": 10.0,
    "leftRightEnabled": true,
    "upDownEnabled": true
},
"sensorialRotation": true, //重力体感控制
"images":
{
    "preload": true,
    "preview": "quitol_p",
    "front": "quitol_f",
    "back": "quitol_b",
    "left": "quitol_l",
    "right": "quitol_r",
    "up": "quitol_u",
    "down": "quitol_d"
},
"camera":             //全景之旅视场参数控制
{
    "keep": "all|~atvRange|~rotation|~fov",
    "atvMin": -90.0,
    "atvMax": 90.0,
    "athMin": -180.0,
    "athMax": 180.0,
    "reverseRotation": true,
    "rotationSensitivity": 30.0,

```



```

        "vLookAt": 0.0,
        "hLookAt": 170.0,
        "zoomLevels": 2,
        "fovMin": 30.0,
        "fovMax": 90.0,
        "fovSensitivity": 30.0,
        "fov": 90.0
    },
    "hotspots": //热点跳转
    [
        {
            "id": 1,
            "atv": 0.0,
            "ath": 170.0,
            "width": 0.05,
            "height": 0.05,
            "image": "hotspot",
            "alpha": 0.8, //跳转过渡透明度
            "overAlpha": 1.0,
            "onClick": "lookAt(0.0, 170.0, true);
load('res://raw/json_cubic_quito2', true, BLEND(2.0, 1.0), 0.0, 190.0)"
        }
    ]
}

```

通过以上 JSON 协议，实现了项目组将程序移植于各类终端的初衷，也为后续开发者继续开发提供了便利。

## 四、系统测试方法

系统测试内容	条件	设备	规则	是否满足
程序是否正常 运行，有无中断	SD 卡访问	华为 P6	安装手机应用	是
用户观看平面 全景图	SD 卡访问	华为 P6	点击 cubic 模式	是
用户观看球形 全景图 2	SD 卡访问	华为 P6	点击 spherical2 模式	是
用户观看球形 全景图 1	SD 卡访问	华为 P6	点击 spherical1 模式	是
用户观看柱形 全景图	SD 卡访问	华为 P6	点击 cylindrical 模 式	是
可移植性测试	1. 平板 2. 苹果手机	1. Nexus 2. Iphone4s	程序下载并运行	是（#4）

测试图：

## 1. 单个全景图



图 3：球形全景图 1



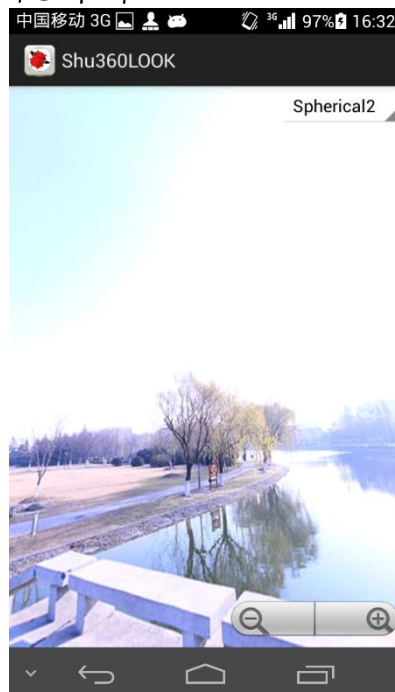
图 4：球形全景图 2



图 5：柱形全景图

## 2. 终端实际效果：

图 6 图书馆图 7 泮池桥图 8 草坪



## 五、结果分析

### 5.1 全景程序结果分析

通过科学的系统测试以及上述效果，本课题组全部完成了全景展示的九个功能，程序很好的支持了三种全景效果(立方体,球形,柱体)，4 种分辨率可以使得用户使用时获得更好的体验。综合来看，本程序的创新之处在于开创了一种用户从没有过的虚拟全景之旅，极大的提高了用户观看相关图片的视角与自主性。因此，与传统图片展示平台相比，我们更注重用户的体验，鼓励他们转动起手机，自主详细的浏览他们想看的图片。

### 5.2 MATLAB 部分

## 六、总结

通过 10 周的忙碌，看到最终的产品呈现在终端产品上，给大家带来不同的观感时，我们小组认为此次项目让我们获益良多。

首先，基于安卓操作系统下的应用开发是现在热门，种类繁多的应用让我们应接不暇。而我们所要开发的这款图像应用在市场上还没有出现。因此，能最终做出来对我们是很大的鼓励。在这个过程中，我系统性的熟悉并学习了关于 android 开发的知识。知识的学习必须经过实践的检验。从最开始提出可行性报告，到基本页面的编写，三种全景图像的嵌入以及各种人机交互功能的实现，我们都一步一个脚印的走过了，经历了很多失败，不断的修改代码，调试以确定最佳的应用品质。尤其是 JSON 协议的加入，使得我们的程序不仅仅局限于安

卓系统阵营，更可以移植到 IOS,WINDOWS 系统中。就目前图像浏览展示的趋势来看，包含更多信息量的全景图不可避免的将代替传统的平面图像。因此，能借公教高的平台实现我们组的目标，对实验室的老师和学校，我们表示由衷的感谢，为我们提供了很多技术支持。