
武汉理工大学毕业设计（论文）

会议白板 APP 的设计与实现

学 院(系): 计算机科学与技术学院

专业班级: 计算机 1404

学生姓名: 鞠 鹏

指导教师: 巩 晶

学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

作者签名：

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障、使用学位论文的规定，同意学校保留并向有关学位论文管理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。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编入有关数据进行检索，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。

本学位论文属于 1、保密口，在 年解密后适用本授权书

2、不保密口。

（请在以上相应方框内打“√”）

作者签名： 年 月 日

导师签名： 年 月 日

摘要

现代的企业会议与当年相比已经截然不同，经济的发展、数字化的推进和科技的创新，对会议提出了更高的要求，需要能在移动设备上使用的会议辅助工具。

所设计的会议白板基于 TCP 协议，利用手机热点进行多用户的连接，可以实现多台 Android 手机同时对白板进行诸如手绘、文字插入、图片插入、撤销和恢复等操作。系统基于 Android studio 进行开发，软件界面简洁友好、操作简单、响应迅速，实现了会议辅助所需的基本功能，并且具有良好的可拓展性和可维护性。

经测试验证，所设计的会议白板可以让现代的企业会议更便捷，沟通更高效。

关键词：Android；会议白板；局域网；多人共享；

Abstract

Modern corporate meetings are completely different from those of the current year. The rapid development of the economy and the rapid advancement of digitalization, as well as the in-depth popularization of cutting-edge smart technologies. There increased demand for modern corporate conference and office.

This software is based on the TCP connection, using mobile phone hotspots for multi-user connections, you can implement multiple Android mobile phones at the same time on the whiteboard for such operations as hand-painting, text insertion, picture insertion, revocation, and recovery. The conference whiteboard designed in this thesis is developed using Android studio. The software interface is simple and friendly, simple in operation, and responsive. It fulfills the basic functions of expected requirements, and has good expandability and maintainability.

After testing, the conference whiteboard based on the Android system can make modern corporate meetings more convenient, communicate more efficiently.

Keywords: Android; Meeting Whiteboard; local area network; Shared by many people;

目录

摘要.....	I
Abstract	II
第一章 绪论.....	1
1.1 研究背景	1
1.2 研究现状	1
1.3 研究意义	2
第二章 系统分析与设计	3
2.1 系统可行性分析.....	3
2.2 需求分析	3
2.3 模块设计	5
第三章 系统实现.....	8
4.1 系统开发技术路线.....	8
4.2 白板编辑模块.....	8
4.3 画笔修改模块.....	9
4.4 撤销反撤销模块.....	10
4.5 插入文字模块.....	11
4.6 插入图片模块.....	12
4.7 白板保存模块.....	13
4.8 白板连接模块.....	14
第四章 系统测试.....	17
4.1 欢迎界面测试.....	17
4.2 白板测试	18
4.3 联机操作测试.....	20
第五章 总结与展望.....	22
5.1 总结	22
5.2 展望	23
参考文献.....	24
致谢.....	25

第一章 绪论

1.1 研究背景

依照 eMarketer 的调查，2016 年全球智能手机用户数量已达 22 亿，其中使用 Android 系统的智能手机的已经达到 86% 以上^[1]。现在的手机厂商越来越多，Android 手机市场巨大，国内像移动、联通这样的网络通信公司已基本定型，使得越来越多的人能够享受网络带来的让人愉悦的精神生活和体会智能手机所带来的方便快捷。

这个时代是 Android 的时代，它凭借入门门槛低、容易学习等优势，迅速得到了开发人员和用户的欢迎，Android 是由 Java 和 Linux 构成的开源软件，因为是开源软件，允许所有厂商和个人在其基础上根据自己的实际情况进行自由开发。因此，它现在成为了世界移动端的主流。

随着网络覆盖范围的不断扩大，越来越多的人开始使用网络在网上获取信息，Wi-Fi、热点等功能应运而生，它们让用户之间的信息传输变得便捷，人们可以更快的浏览和获取信息。

手机和网络带来的便利，让现代的人们开始不再动笔写字，笔和纸的使用越来越少，人们更加倾向于享受手机、平板或者 PC 等电子设备给自己带来的便利，企业会议中有很多事情用语言描述起来并不方便，但会议白板有效的解决了这一问题。

1.2 研究现状

上个世纪，电子白板在欧美开始兴起，在本世纪初已经具备一定的市场，并且发展迅速，接下来的几年，在政策驱动下，电子白板市场依旧在继续扩大。市场增大导致厂商也增多，他们之间的竞争也很激烈。中国的电子白板发展起步比起欧美是要晚了不少，但是增长也十分迅速。

国内外电子白板产业目前在教育领域也已经有了作为，android 技术的发展在教育领域也产生了深远影响，教育市场目前正在蓬勃发展，例如院刚在《Android 端课堂交互应用研究与实现》中所提到的基于 Android 的课堂交互应用软件，他采用了基于游戏绘图的 surfaceview 来实现，通过这种方式，对手写绘图进行了优化^[3]，而工业和信息化教育上刊载的《Android 移动客户端同步手绘板的设计与实现》将目标转向儿童，他实现了手机和 PC 端之间的同步手绘板，儿童通过 Android 手机端进行绘图，同时 PC 端会显示手机端绘制的内容，这样的软件对于孩子的动手能力和智力开发都是很有好处，还能提高孩子的绘画兴趣^[5]。

与此同时，国内外在手写识别方面的研究也有了很大的跨越，准确率和识别速度都有了新的创新和突破，何聪在《基于 Google Android 和云计算的手写识别系统的研究与视线

中》采用了云服务器技术来提高系统性能，并没有照搬传统移动手机手写应用的思想，而是另辟蹊径，为未来的手写识别技术提供了新的思路，但他的整个工作目前还处在探索阶段^[4]，何聪等人在《基于云计算平台的手写识别系统》中采用了类似的技术，这里就不过多赘述，但是朱虎明等人在《Android 平台上基于卷积神经网络的手写数字识别实验》中，让我了解了有关卷积神经网络方面的内容，它可以用来识别扭曲不变形、位移和缩放的二维图像^[7]，他们开创了一种新型的服务应用模式，将 Android 的手写输入推向以网络为载体、以服务为核心的新模式。研究者们都在为此付出努力，不难想象，将来的手写操作一定会更加便捷。

1.3 研究意义

现代的企业都要求高效的沟通，如果能对会议室的办公设备进行智能化升级则正好可以满足企业的需要。以前的会议室，大多使用的还是投影白板，但会议投影白板时代已经落幕。

本次毕业设计的研究目标是开发出一款基于 Android 的开会辅助工具——会议白板 APP。在同一个会议室里的人在各自的 Android 手机上操作各自的白板，与会者可以在白板上任意绘制，其他与会者能够同步地看到其他人绘制的内容。该系统支持多人同时参与；支持文字输入；支持手绘功能；支持插入图片；支持橡皮擦功能；支持撤销、反撤销等操作。

现在移动手机相当普及，不分男女老少，人们都可以拥有自己的手机，网络相较以前也有了很大的提高，网络的搭建非常简单，几乎所有的手机都拥有 Wi-Fi 连接和热点功能，为本 APP 的实现和发展提供了可能。通过本 APP 能够让现代的企业会议更便捷，沟通更高效，这样本研究就有了进行的意义。

第二章 系统分析与设计

2.1 系统可行性分析

可行性分析就是确定是否能够设计出这样一个系统。本系统的可行性分析从以下几个方面入手：

1) 经济可行性：在开发会议白板 APP 时，主要使用 Android Studio 来进行开发，对环境的搭建也没有特殊的要求，因为不需要大规模的计算和数据的读取，所以对手机的性能要求也没那么高，Google 为我们提供了丰富的 API 类库，并且网上也有各种开源代码可以供我们选择下载参考，还不需要支付任何费用，所以不用考虑到经济问题。

2) 技术可行性：本系统在 Android 的环境下用 Java 语言编写，技术简单而且可靠。Java 语言的两大特征是功能强大和简单易用。TCP 协议的技术到了现在也已经发展的非常成熟，想获取的知识在网上都能找到，当前的计算机硬件配置也能满足本系统的开发需求。所以该系统在技术上是可行的。

3) 运行可行性：该系统从现在社会发展需求出发，了解了当今形势对会议白板这方面确实有需要，而且 APP 的安装和使用对于现在的用户来说非常简单，容易推广。所以从运行角度上看，该系统是可行的。

从经济可行性，技术可行性，运行可行性三方面入手，确定该系统的设计与实现是可行的。

2.2 需求分析

本款基于 Android 系统的会议白板能够实现多名用户同时参与，并且在使用过程中不会出现较大的卡顿，参与会议的人员是本软件主要的作用范围，通过会议白板软件，在同一个局域网的人员在各自的手机上绘制。当某一与会者在白板上绘制时，其他使用者也能同步的看到绘制的内容，当小组会议或者其他会议需要讨论问题时，本软件就可以排上它的用场。同时因为本软件需要使用手机的热点来建立局域网连接，所以所有参与会议的人员的物理位置不能相距太远，不然信号不好网络较差数据传输效果也会很差，如果局限在比较小的范围内，使用效果就会好很多，如同一间教室或者会议室都可以。

为更好的描述用户需求，以及描述该系统所实现的功能，绘制了该系统的用例图，如下图 2.1 所示：

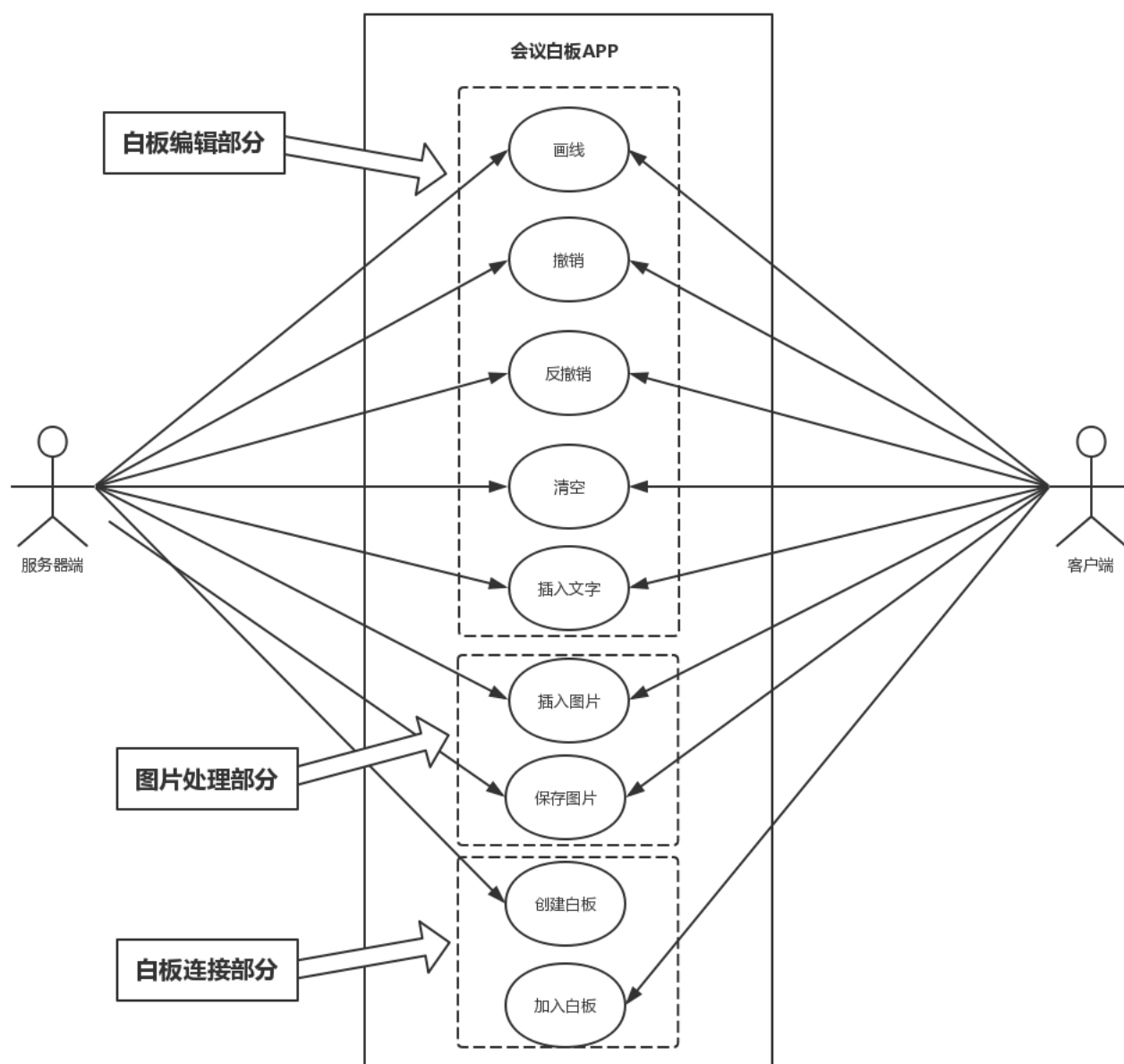


图 2.1 会议白板 APP 用例图

会议白板各功能的具体描述如下：

首先是白板编辑部分：

- (1) 用户能在屏幕上随意的涂画，可以进行画笔颜色、画笔宽度、画笔种类的修改，本软件提供了红绿蓝三种颜色、细中粗三种宽度、浮雕和模糊两种类型的画笔供用户自由搭配使用。
- (2) 能够对编辑过的路径进行撤销和反撤销操作，撤销可删除最近一次操作，反撤销可恢复被删除的操作。也可以选择清空白板，清空白板的同时也会清空所有

路径数据。

- （3）能够在任意自己想要的地方插入文字，文字和画笔一样，同样可以选择三种颜色和三种文字大小。
- （4）能够选择橡皮擦工具，对不想要的部分进行擦除。

其次是图片处理部分：

- （1）能够在白板上插入图片，图片可以从本地相册中任意选择，同时画笔、橡皮擦等工具也都可以对图片进行修改。
- （2）能够对编辑过的白板进行保存操作，将编辑过的白板以图片的方式保存到本地上方便用户之后查看。

最后是白板连接部分：

- （1）无网络情况下的单人白板：在无网络的情况下，可以打开单人白板进行自由创作，随时记录并保存信息。
- （2）局域网下使用会议白板：双方处在同一局域网时，主机开启手机热点，然后新建白板，其他人连接热点后输入主机 IP 连接进入即可共同编辑白板。

2.3 模块设计

本系统设计的会议白板 APP 主要功能模块包括：白板编辑模块、画笔修改模块、撤销反撤销模块、插入文字模块、插入图片模块、白板保存模块、白板连接模块。用户根据自己需要选择单人操作或者连接他人热点进行联机操作。连接成功后，可与另一方快速，方便的进行信息交换和问题讨论。打开 APP，进入到初始界面，在用户加入编辑成功后，即可开始绘制等操作，也可以通过返回按钮进行返回初始界面的操作。在用户输入对应主机的 IP 进行连接时，会进行判断，如果连接失败则自动进入个人画板编辑模式，其中支持的基本画板操作有：画笔类型修改、画笔颜色修改、画笔宽度修改、插入文字、插入图片、撤销和反撤销，保存白板等操作。软件功能模块图如图 2.2 所示。

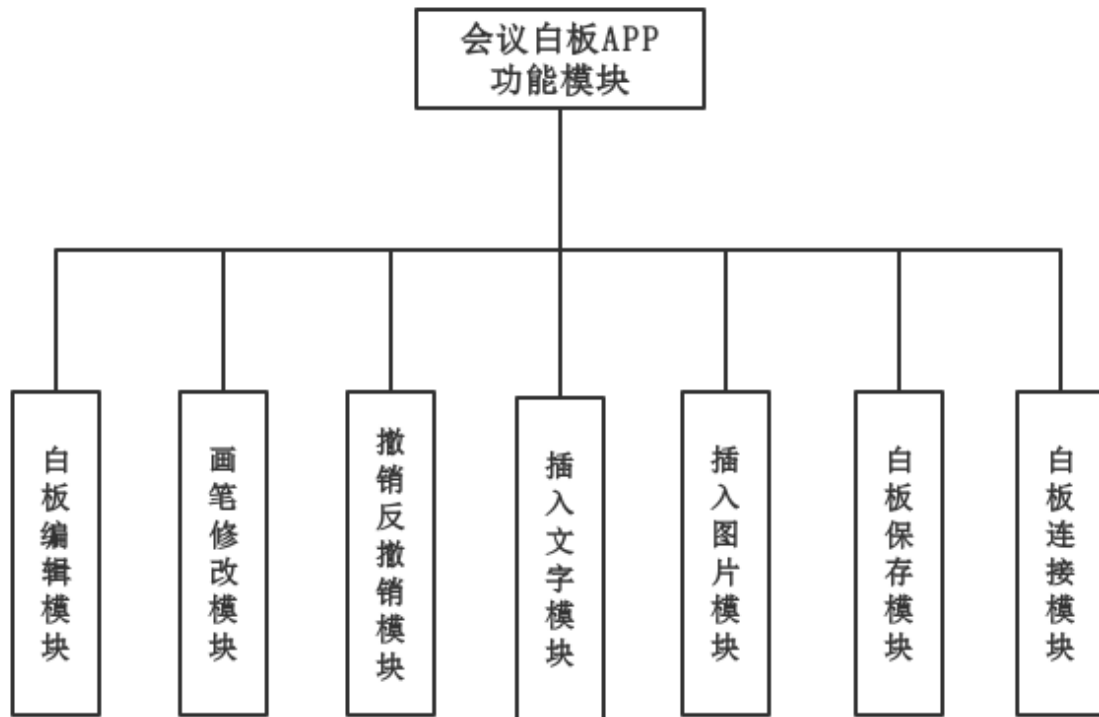


图 2.2 软件功能模块图

上述模块中最难实现的是白板连接模块，白板连接模块有几个难点，在后面会重点描述，因为是局域网，所以采用的是 socket 连接，软件实时的接收和发送数据，并将白板更新，以使会议白板具有实时性。

接下来对图 2.2 各功能进行简单的描述

- 1) 白板编辑模块：在白板界面，可以随便使用手指或者触笔进行绘制，多种画笔种类和颜色粗细的搭配也满足了用户会议讲解的需要，而且由于我将功能键全部放在菜单栏里也使得白板界面完全放空，给了用户充分的空间用来绘制，也不用担心误操作的产生，可以有效提高会议效率。
- 2) 画笔修改模块：我将橡皮归为特殊的画笔，所以用户可以在画笔种类里进行选择，既可以选模糊类型，也可以选带有阴影的浮雕类型，颜色和画笔宽度方面，我设计的不够到位，应该做的更加对客户友好些，提供更多选择会更好。
- 3) 撤销反撤销模块：撤销功能就是删除自己上一步的操作，由于软件总体偏小，所以就没限制撤销的次数上限，这样也方便了用户多次撤销也不用担心由于次数的限制导致有些操作被“钉死”在白板上而无法操作。反撤销就是将删除的操作重新添加到白板

上，可以让用户更灵活的对白板进行编辑。

- 4) 插入文字模块: 根据用户需要在用户想要的地方插入文字, 因为手写字有时并不好用, 而且也不容易辨认, 这时候插入文字就可以更直观、更清晰显示用户想要表达的内容。
- 5) 插入图片模块: 点击插入图片后, 打开本机的相册, 可以选择任何想要显示的照片添加到白板上进行操作, 可以对图片进行画笔标记或者橡皮擦除等操作, 同时也兼容了 **Android** 的版本, 4.4 版本前后对手机相册图片的处理方式有所不同, 也为了防止出现卡顿, 对于过大的图片, 在插入白板之前会先进行压缩处理。
- 6) 白板保存模块: 点击保存图片后, 即可将当前白板的内容保存在本地地上。
- 7) 白板连接模块: 开启热点的手机点击新建白板, 即可新建服务器并打开白板界面, 其他连接到热点的手机, 在输入热点手机的 **IP** 后, 点击加入编辑, 连接成功即可进入多人编辑模式, 连接失败则会进入单人编辑模式。

第三章 系统实现

4.1 系统开发技术路线

本会议白板 APP 运行在 Android 手机上，但不只是为了会议服务，所以也分为了有网和无网的情况。只是为了记录信息的时候，不需要联网，直接开启单机模式，当要和其他人建立会议连接的时候，则需要一台手机开启 Wi-Fi 热点来创建局域网。

《Android 学习之绘图入门》中对绘图的基础知识进行了介绍，Android 的绘图编码应该继承 View 组件，并重写他的 onDraw (Canvas canvas) 方法即可重写 onDraw (Canvas canvas) 方法时涉及一个绘图 API: Canvas, Canvas 代表了“依附”于指定 View 的画布。”它提供了很多方法供我们使用。

局域网下白板连接：Android Studio 里的 JDK 的 java.net 开发包中提供了 socket 的相关类，只要主机打开手机的热点，创建局域网络，另一方联入该热点，双方即处在同一局域网中。在三次握手后双方建立连接，即可进行数据的交换。

4.2 白板编辑模块

白板编辑实现较为简单，重点是使用双缓冲技术来实现画图，当数据量较大时，绘图可能需要较长时间，而且有时还会出现闪烁现象，双缓冲技术就是为了解决这个问题而被采用的。双缓冲就是在内存中创建一个与屏幕绘图区域一致的对象，先将图形绘制到内存中的这个对象上，再一次性将这个对象拷贝到屏幕上，这样会大大加快绘图速度。

双缓冲的实现过程如下：

- 1、在内存中创建与画布一致的缓冲区
- 2、在缓冲区画图
- 3、将缓冲区位图拷贝到当前画布上
- 4、释放内存缓冲区

代码中的实现：定义一个内存中的图片 cacheBitmap，该图片将作为缓冲区，再定义 cacheBitmap 上的 canvas 对象，设置 cacheCanvas 将会绘制到内存中的 cacheBitmap 上，最后在重写的 onDraw 方法里，将 cacheBitmap 绘制到 View 组件上。

双缓冲示意图如图 4.1 所示：

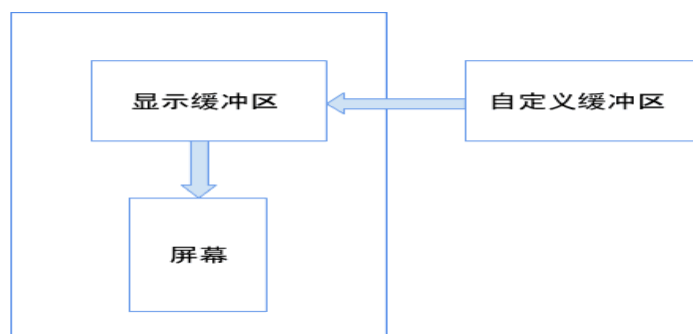


图 4.1 双缓冲示意图

4.3 画笔修改模块

Android 自带的 `paint` 类型拥有很多属性可以设置，画笔修改也就是调用相应的方法实现 `paint` 属性的修改。

画笔的 `style` 有三种：`Paint.Style.STROKE` 只绘制图形轮廓、`Paint.Style.FILL` 只绘制图形内容、`Paint.Style.FILL_AND_STROKE` 既绘制轮廓也绘制内容。但是对于本程序来说，三种 `style` 其实并没有什么太大区别。再就是使用 `setAntiAlias` 方法抗锯齿，使用 `setDither` 方法防抖动。

模糊类型和浮雕类型稍微有些特别，但本质也是 `paint` 的一个属性，`setMaskFilter(blur)` 方法用来设置模糊属性，`blur` 参数包含度数和样式。而 `setMaskFileter(emboss)` 方法用来设置浮雕类型，但是 `emboss` 的参数就复杂很多，`direction` 是 `float` 数组，定义长度为 3 的数组标量 `[x,y,z]`，来指定光源的方向；`ambient` 取值在 0 到 1 之间，定义背景光 或者说是周围光；`specular` 定义镜面反射系数；`blurRadius` 为模糊半径。

`setColor` 用来设置颜色，`setStrokeWidth` 和 `setTextSize` 则分别对应了画笔的粗细和插入文字的大小。

由于我将橡皮擦也划在了画笔种类中，但其实橡皮擦功能的实现确实体现了这一情况，Android 非常人性化的设计了 `setAlpha` 和 `setXfermode` 方法，再只需要将画笔颜色设置为 `TRANSPARENT` 也就是透明就可以实现橡皮擦的功能。

关于 `setXfermode` 的参数，使用图片来解释更加清晰，参见下图 4.2



图 4.2 图像合成效果示意图

4.4 撤销反撤销模块

撤销和反撤销实现的前提，就是要将每次绘制的路径存入栈中，本程序就是存入 List 中。

初始化两个 List，路径保存列表 `savePath` 和路径删除列表 `deletePath`，分别用来保存绘制的 path 路径和撤销掉的 path 路径

撤销功能步骤：

- 1) 调用初始化画布函数将画布清空。
- 2) 将路径保存列表 `savePath` 中的最后一个路径保存到路径删除列表 `deletePath` 中，然后将这条路径从路径保存列表 `savePath` 中删除。
- 3) 取出 `savePath` 中的所有路径，重绘在画布上面。

反撤销功能步骤：

- 1) 将路径删除列表 `deletePath` 中的最后一个，也就是最顶端路径取出栈，并且加入到路径保存列表 `savePath` 中。
- 2) 将取出的路径重绘在画布上。
- 3) 从 `deletePath` 中删除最后一个路径。

思路很好理解，操作起来也不难，重点是要将画布上绘制的路径保存在列表中，也就是在按下时，重新创建路径 `path`，抬起时，将按下时创建的路径设置为 `null`，这样才能依

次保存每个路径。

再次基础上，可以顺便实现画笔的清空功能：

1) 直接调用画布的初始化操作清空画布。

2) 将路径保存列表 `savePath` 和路径删除列表 `deletePath` 清空
撤销操作流程如图 4.3 所示，反撤销类似所以不再贴出图片。

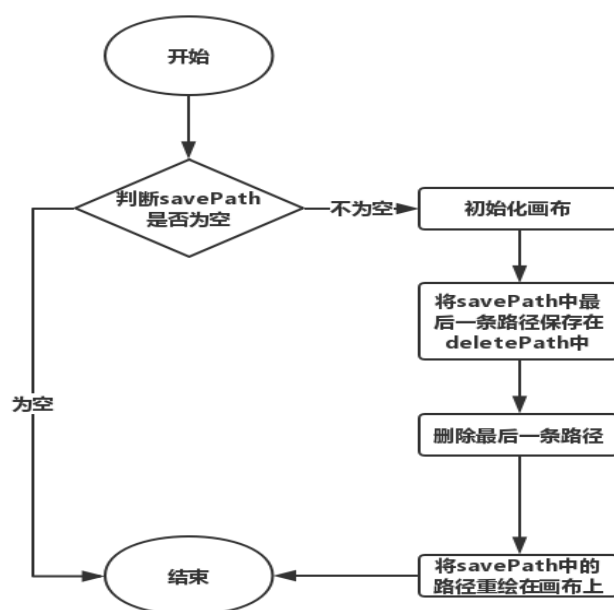


图 4.3 图像合成效果示意图

4.5 插入文字模块

插入文字功能的实现思路并不难，但是在实现过程中出现的问题很多。首先设置一个弹框 `PopupWindow` 进行文本的输入，在弹框里面的空间就是一个 `EditText`。当点击白板后，弹框 `PopupWindow` 出现，软键盘也自动弹出，用户输入文字，再点击屏幕任意位置后，弹框消失，同时获得弹框中输入的文字，再次点击想要显示文字的地方区域时，即可插入文字。

前面提到画笔 `style` 的时候说了有三种，在显示文字的时候就要将画笔的 `style` 修改为 `FILL`，如果是 `STROKE`，那么文字就会被绘制成空心文字，由于加入了插入文字功能，所以前面定义的 `DrawPath` 类和撤销反撤销方法也需要对应修改，增加了用来保存文本的 `mText` 和保存插入位置的 `mOffX` 和 `mOffY`，`ACTION_DOWN` 时也增加了一次判断，如果当前选择的是画笔则进行正常绘图操作，如果是插入文字则弹出弹框 `PopupWindow`。

撤销和反撤销方法中也同样需要增加一个判断，判断当前列表里最后的一个元素的 `mText` 是否为空。如果不为空，就要对文字进行操作。

插入文字相较于前面的模块要复杂些，所以这里用流程图进行解释。

点击插入文字选项后，用户在屏幕的任意位置点击即可显示出键盘，输入想要输入的文字后，点击屏幕你想要文字显示的位置即可将文字插入。流程图如图 4.4 所示：

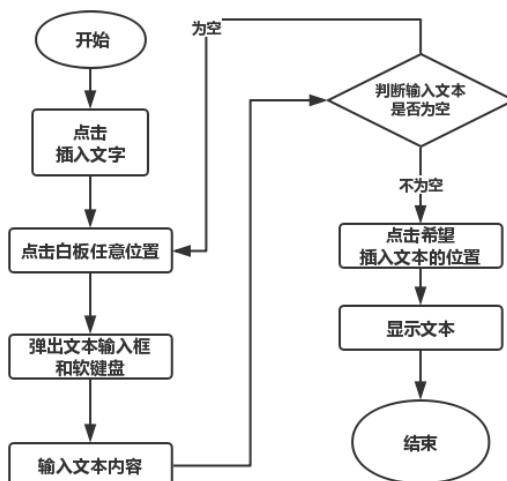


图 4.4 插入文字流程图

4.6 插入图片模块

由于需要进行读写操作，首先要在 `AndroidManifest` 中声明权限，调用系统相册后，进行图片的选择，之后需要判断手机系统的版本号，4.4 及以上系统需要先对 `Uri` 的类型进行判断，如果是 `document` 类型的 `Uri`，则需要通过 `document id` 处理，如果不是 `document` 类型的 `Uri`，则使用普通方式处理，但是 4.4 以下的系统就不需要进行 `Uri` 类型的判断，真实的图片路径是通过 `Uri` 和 `selection` 来获取。

之所以要这么做，是因为 `Android` 系统从 4.4 版本开始，选取相册的图片不再返回图片真实的 `Uri`，而是一个封装过的 `Uri`，因此如果是 4.4 版本以上的手机需要对这个 `Uri` 进行解析才行。

获取到图片后，需要考虑对图片进行压缩。循环判断图片是否大于 300KB，如果大于就采用质量压缩，压缩后还要根据图片的宽高进行缩放，为了防止图片显示后占满整个屏幕。对图像的这部分操作封装成一个 `ImgUtil` 类，便于后面的使用。

选择插入图片后，会打开手机相册列表，在相册文件中选择想要插入的图片，成功获取到图片后根据 `Android` 系统版本判断如何处理图片，再根据图片大小判断是否需要压缩，然后将图片显示到画板上。流程图如 4.5 所示

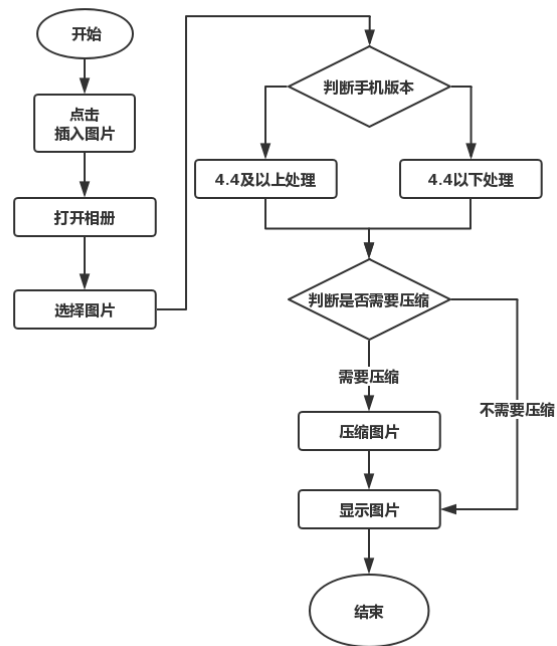


图 4.5 插入图片流程图

4.7 白板保存模块

白板保存相对简单，先获取系统当前时间，并以该时间作为文件名，这样做的原因是为了防止重复保存。由于之前白板绘制部分的实现，我利用了 `cacheBitmap` 和 `CacheCanvas` 的双缓冲来进行绘制，所以我直接将 `cacheBitmap` 保存为图片即可。

保存图片序列图如下图 4.6 所示：

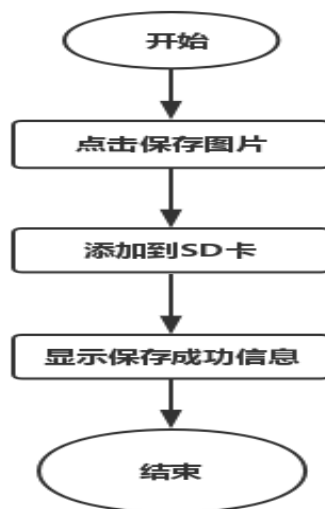


图 4.6 保存图片序列图

4.8 白板连接模块

白板连接模块是本软件中最重要、也是最难实现的一个模块。JDK 中的 `java.net` 开发包中提供了 `socket` 的相关类，双方需要处在同一局域网中。1) 文件传输方或者接受方打开手机的热点，创建局域网络，另一方联入该热点，双方即处在同一局域网中。2) 双方连接同一路由器的无线网也处在同一局域网中。文件发送方向接收方发送请求，请求建立连接，在三次握手后双方建立连接，发送方可通过 `socket` 发送文件数据。

`socket` 也叫套接字，是一个对 TCP/IP 协议进行封装的编程调用接口（API），通过 `Socket`，我们才能在 Android 平台上通过 TCP/IP 协议进行开发。

`socket`（套接字）的连接过程分为以下三个步骤：

1) 服务器监听：服务器端的套接字处于等待状态，不会确定连接某一个具体的客户端套接字，它所做的事情是实时监控网络状态，等待客户端的连接请求。

2) 客户端请求：客户端的套接字向服务器端的套接字提出请求连接。在此过程中，客户端的套接字做了两个工作，第一个是确定连接目标，描述连接目标即服务器端套接字（IP 地址和端口号），第二个工作是向连接目标提出连接请求。

3) 确认连接：服务器端的套接字处于实时监控状态，当客户端套接字发出连接请求时，服务器端的套接字接收到该请求，开始对这个请求进行回应。建立一个新的线程来把服务器端的套接字描述（IP 地址和端口号）发送给客户端进行确认。客户端确认此描述后双方就成功建立连接。此时服务器端继续接受其他套接字请求，不受影响。

`socket` 通信模型如下图 4.7 所示

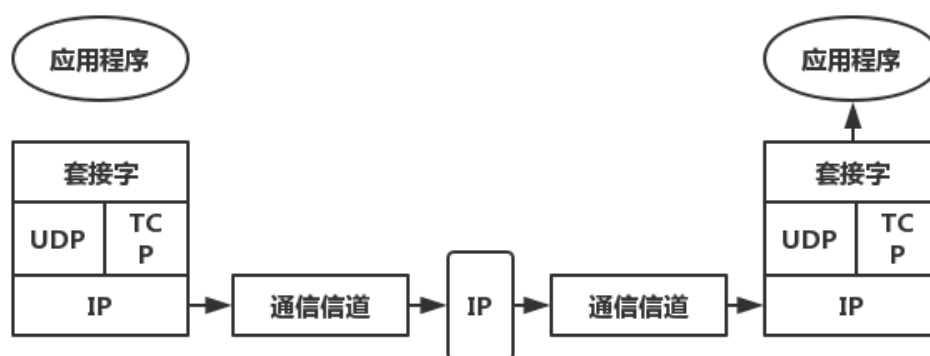


图 4.7 socket 通信模型

在应用程序两端连接成功的时候会产生一个 `socket` 实例，接下来就可以通过操作这个实例来进行白板数据传递操作。

`socket` 连接代码过程图如图 4.8 所示：

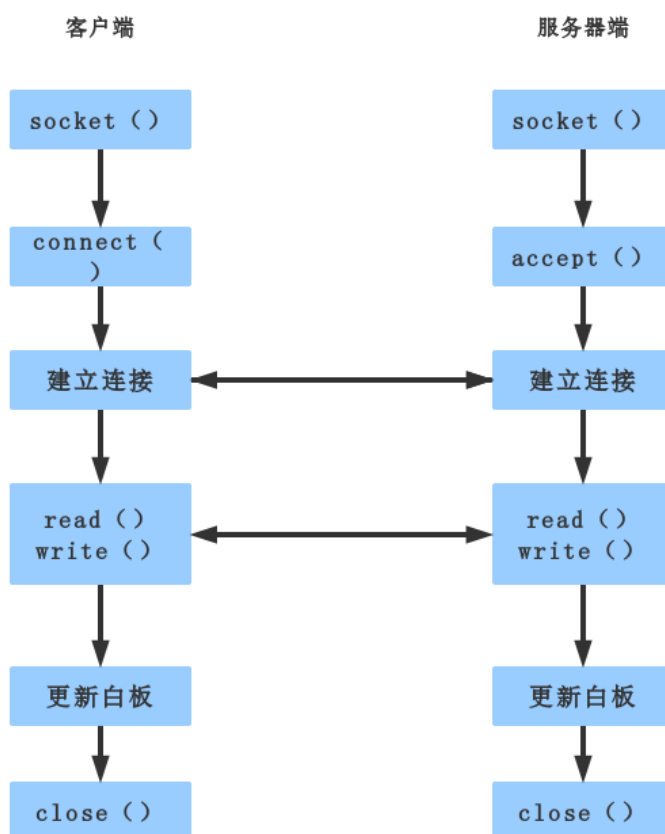


图 4.8 socket 连接过程图

首先由主机开启 Wi-Fi 热点，然后点击新建白板，其他与会者连接热点后，手动输入主机的 IP 地址，点击加入编辑，系统进行判断，如果 IP 与主机不符合则进入单人编辑模式，点击返回键后可重新输入 IP，如果 IP 符合即可加入主机白板，进入多人编辑模式。流程图如 4.9 所示

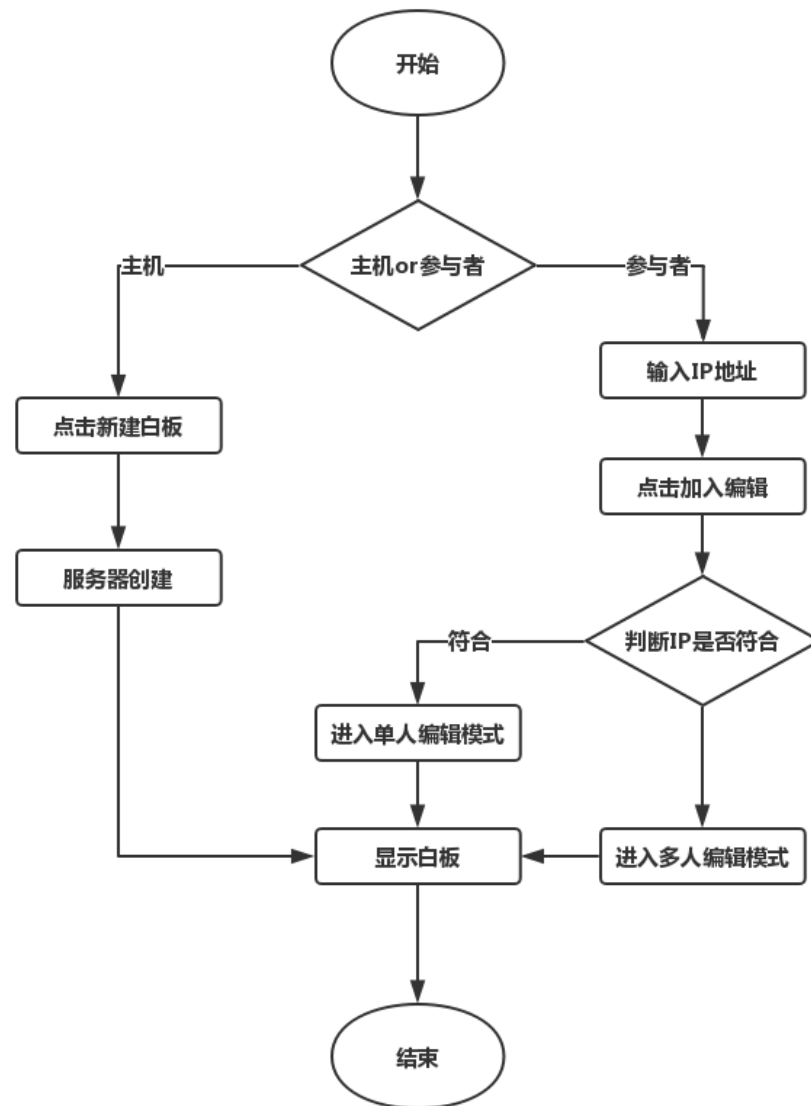


图 4.9 白板连接流程图

第四章 系统测试

4.1 欢迎界面测试

现在的 APP 打开都应该要有一个欢迎界面，欢迎界面的作用很多，比如可以贴出广告作为额外收入，或者用来预加载系统数据，又或者是用来提高用户的兴趣。

欢迎界面 3 秒后，进入初始界面，为了方便使用，初始界面我做的非常简洁，也没有添加任何多余的东西，为的就是希望能够提高用户体验，点击新建白板后，打开白板，同时本机就会作为服务器端，需要开启热点供其他用户连接，同时接受其他用户的数据并转发给其他用户。

如果想要连接服务器端，就需要输入服务器端的 IP，然后点击加入编辑，如果 IP 输入没有错误，就能成功连接服务器，如果输入错误，则连接失败，进入单人白板编辑模式。

这里稍微不够人性化的地方就是 IP 需要用户自己手动输入，之后如果有优化的机会，可以将输入变成搜索当前可选 IP，这样会更加方便。

在此基础上，还可以增加房间这个概念，可以多个房间，各自编辑各自的白板，功能就更加丰富了。

下图 5.1 为欢迎界面，图 5.2 为初始界面：



图 5.1 欢迎界面



图 5.2 初始界面

4.2 白板测试

打开白板后，整个屏幕都是白板的界面，所有功能我都选择放在菜单栏里，这样做的好处就是使界面足够简洁，同时不会因为绘制过程碰到哪个按钮而导致误操作，实现的功能下图 5.3 和图 5.4 所示：



图 5.3 菜单栏的显示方式（1）

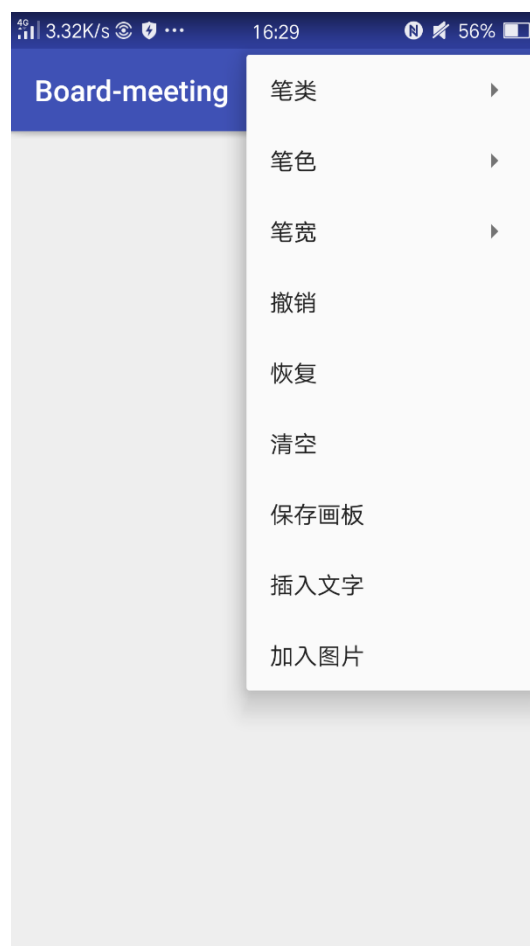


图 5.4 菜单栏的显示方式（2）

首先是画笔的测试部分，在笔类里我提供了两种画笔的使用，同时橡皮擦功能也归类在里面。画笔颜色我提供了红绿蓝三种，后续可能的优化就是增加调色盘之类的设计，可能根据用户自己的需要选择自己喜欢的颜色。画笔宽度同理，我只是简单的设计了细中粗三种，我也可以优化为笔宽的自定义设计，用户可以根据自己情况进行修改，这都是以后可以修改的地方。

下图 5.5 为画笔样式展示，图 5.6 为插入文字测试，图 5.7 和图 5.8 分别为撤回和恢复测试。

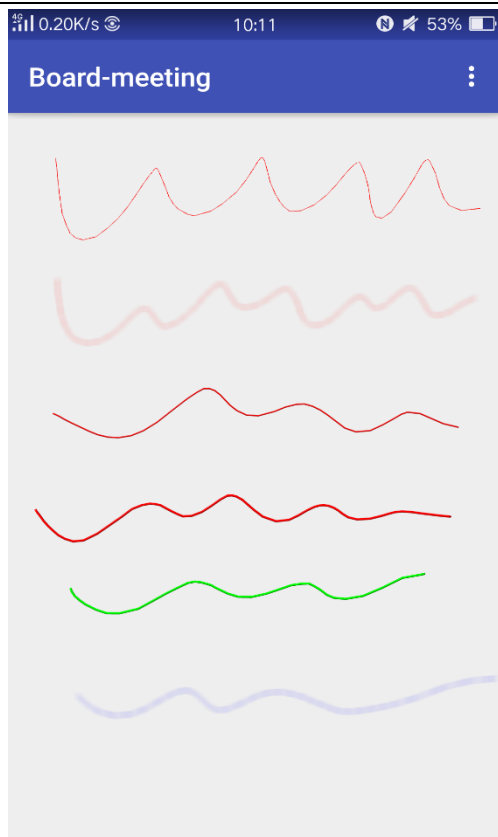


图 5.5 画笔样式

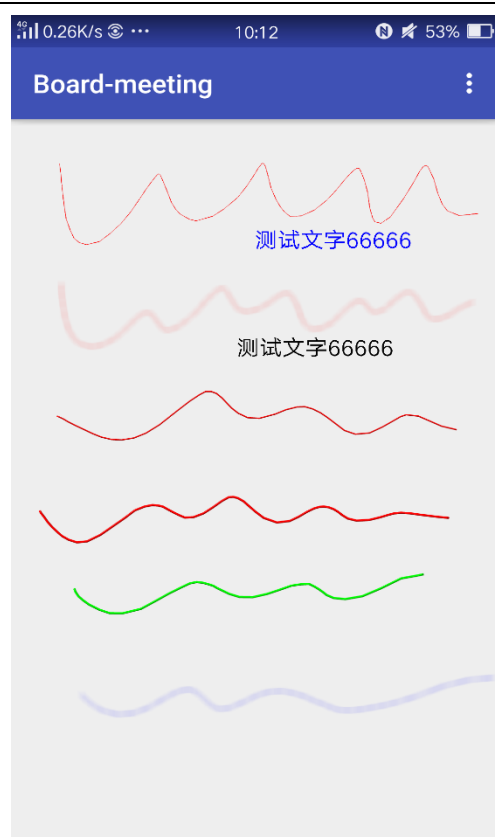


图 5.6 插入文字功能

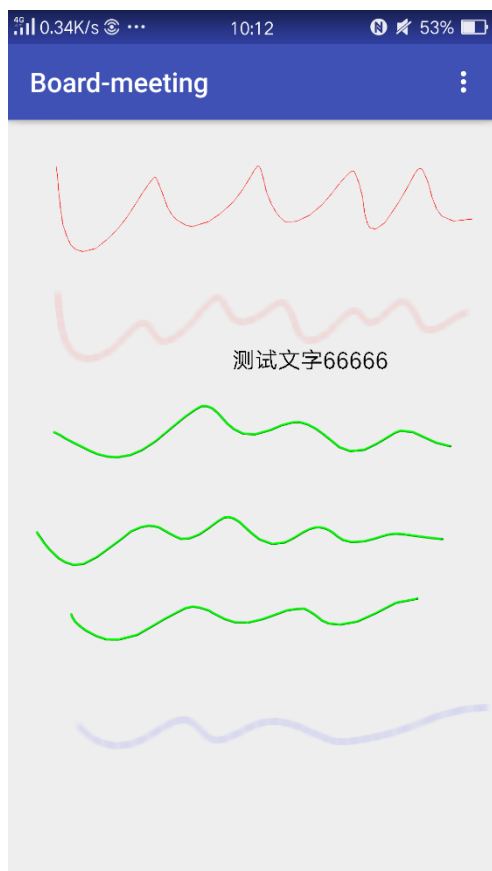


图 5.7 撤销测试



图 5.8 恢复测试

插入图片并没有实现对图片的大小修改和拖动功能，一旦选择只能显示在屏幕的左上角，虽然在显示之前对图片文件的大小进行的判断和压缩，但在实际显示上并没有太多差别。

保存图片功能没有问题。

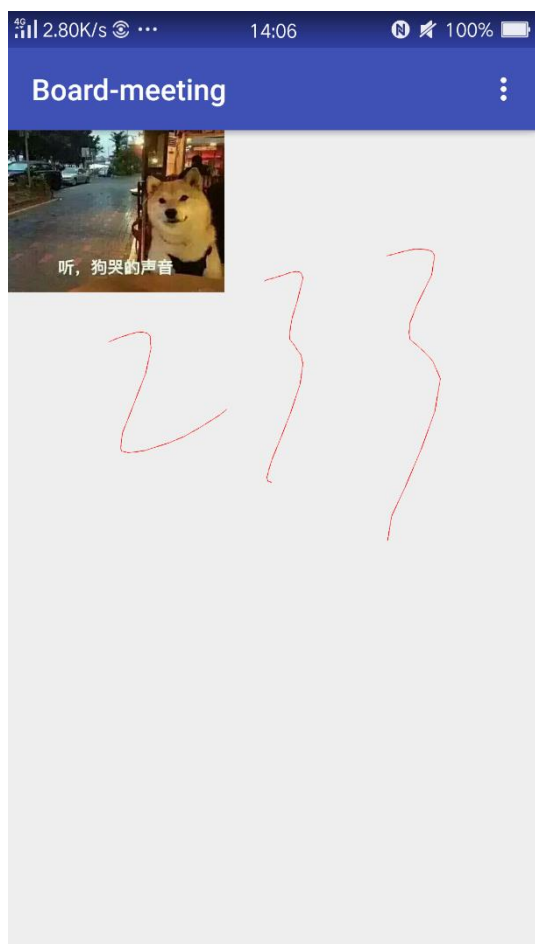


图 5.9 插入图片测试

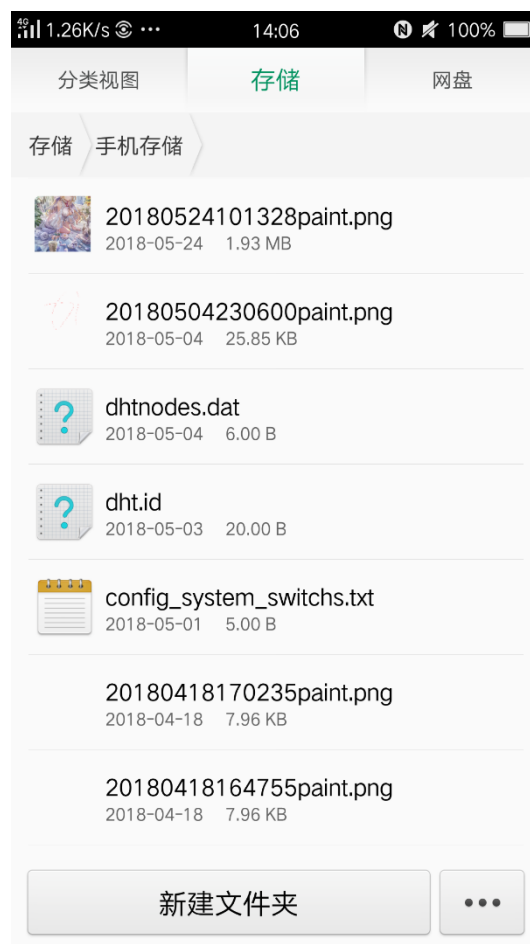


图 5.8 保存白板测试

4.3 联机操作测试

要做服务器端的手机打开热点，同时让其他手机连上热点，服务器端手机点击新建白板创建服务器，其他手机输入服务器端手机的 IP 点击加入编辑。

连接成功后即可进行各自的操作。

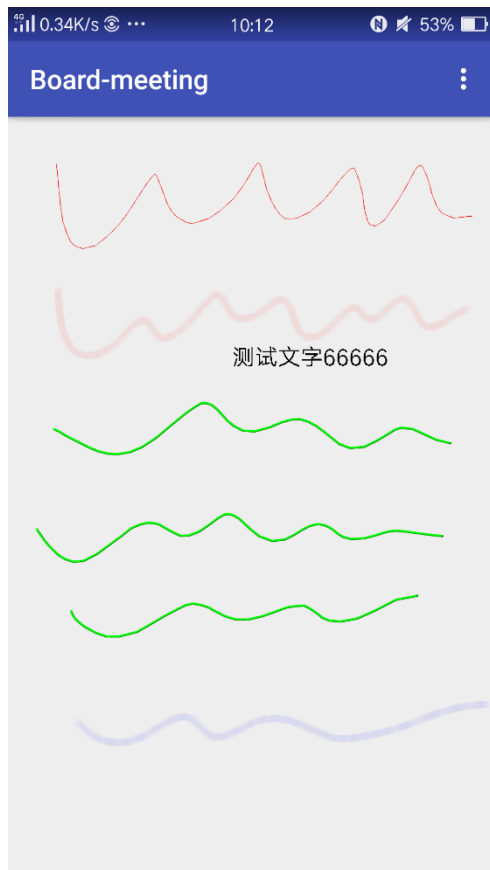


图 5.9 联机测试（1）

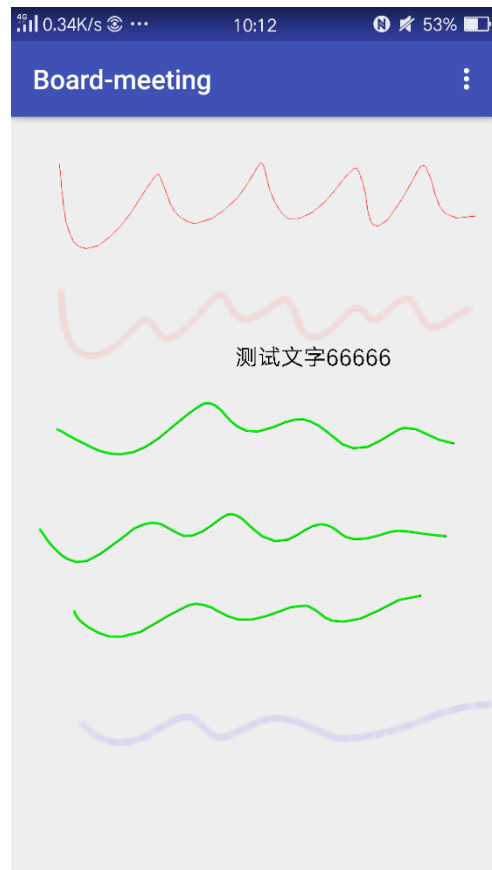


图 5.10 联机测试（2）

在测试过程中，发现偶尔会出现数据丢失的情况，表现为本来应该同时插入文字，但接收端并未接收到文字信息，导致无法显示的问题，需要思考如何改进，但其他方面测试较为成功，画笔的颜色种类粗细等属性，均可同步，路径也能按照需求同时显示。

第五章 总结与展望

5.1 总结

本系统是一个基于 Android 的局域网下的会议白板 APP，该白板作为一个小型会议来使用基本没有什么问题，能够满足会议中用户之间正常沟通和交流的需要，也基本实现了各种基本功能，但是仍有很多不足之处，由于时间原因，暂时没法优化。

截止到目前为止，会议白板 APP 系统的开发基本结束了。内心还是有很多想感慨的地方，选题的时候我面对这个题目并不是那么信心十足，但也认为应该不是个很难的题目，但实际开始开发后，问题接踵而至，这也是因为我在此之前对于 android 方面的知识了解的并不够，同时由于 android 开发是基于 java 语言的，也考验了我 java 语言的掌握程度，所以我要想完成这个系统，要下的功夫一定得多。但是换个角度思考，虽然这个题目对我来说有些难度，但我在做的过程中，同样可以学到很多很多东西，况且 android 开发现在在网上也有很多资源可以让我利用，趁着这个机会增强自己的实力和理论基础，我真的可以收获很多。

系统设计初期，我做了不少准备工作，比如去图书馆寻找相关书籍，去论坛浏览相关帖子，虽然我选过移动应用开发的专业选修课，但是学完之后我的掌握情况并不理想，最多只能算是有所了解的程度，所以我一度十分担心，不知道该怎么做，只能照着书上一点点模仿，很多时候以为自己弄懂了，实际动手还是什么都没学会，感谢在这段时间同学和老师给予我的帮助，让我认清了方向，开始一点一点实现系统功能。

系统设计中期就是在边做边学，遇到了各种各样的问题，也尝试了各种各样的办法去解决问题，在开题报告阶段，由于自己的认识不足，认为通过某种方式就能实现这个功能，然而在实际编程的过程中，经常就会发现，原来自己的想的根本就行不通或者是有很大问题，没办法只能重新开始查找资料，遇到自己怎么也解决不了的问题，也开始尝试着在网上发帖询问，虽然并不是每个人的回复都能帮我解决问题，但是他们的回复让我感受到了我们这个团体的一种积极的氛围，他们会鼓励我继续学习或者是帮我列出类似的知识点或是给出他们自己的想法，这段时间，我学到了很多。

系统设计后期就是对项目的完善和测试过程，虽然大体功能实现了，但是各种各样的 bug 还是很多，也有很多虽然现在没发现，但是用起来或许会出现的隐患，这段时间与前段时间的为了实现功能而想尽一切办法不同，在完善的过程中，会深刻的认识到，自己的编码习惯有多差，修改起来有多麻烦，命名也随意确定，注释也是随心所欲，虽然不是让别人来看，但是自己看起来都很麻烦，为了修改一个问题，可能整个程序都要跟着一起改，这也是我必须引以为戒的。

通过这次毕设，我感觉我收获了很多，也深刻的认识到了自己的不足，同时为自己大

学几年时间的怠惰自责，不过这已经是没办法改变的事，接下来的时间我真的应该静下心来好好学点知识了。

5.2 展望

目前会议白板经过测试后，在多部基于 Android 系统的手机上可以正常使用，但仍有很多不足之处，可以有更多的发展空间。

首先，我只测试了几部手机，并不能保证所有机型都能正常使用，各个厂商所设计的手机都有其独到之处，虽然这样让用户有了更多的可选空间，但同时也对开发者们提出了挑战，因此，开发软件的同时也需要注意考虑系统的兼容性。

其次，在白板连接方面，数据的传输并不精确，偶尔会出现数据漏传的问题，这个问题由于时间原因，我暂时没能解决，同时我只实现了热点所创建的局域网下的用户连接功能，对于现代的环境来说，我可以在次基础上增加 Wi-Fi 连接下的用户连接功能，这样可以给用户更多的选择，也能让用户使用起来更加便利。

最后，我对于撤销和反撤销功能，并没有设计同步操作，也就是用户使用撤销功能后，其他用户并不会同时撤销，这也是由于我个人原因没能实现完全。

虽然本次毕业设计做出的成果不是那么完美，但未来可以发展的空间很大，只要再次基础上加以改进，将来一定能对更多的用户提供更好的帮助。

参考文献

- [1] 申晋祥,鲍美英.基于 Android 的便签 APP 的设计与实现[J].山西大同大学学报(自然科学版),2017,33(06):9-11.
- [2] Android 编程权威指南（第 3 版），比尔·菲利普斯，人民邮电出版社，2017-06-01
- [3] 院刚. Android 端课堂交互应用研究与实现[D].华中师范大学,2015.
- [4] 何聪. 基于 Google Android 和云计算的手写识别系统的研究与实现[D].华南理工大学,2011
- [5] Android 移动客户端同步手绘板的设计与实现[A]. 工业和信息化教育. 2095-5065（2017）12-0074-04
- [6] 何聪,金连文,周贵斌.基于云计算平台的手写识别系统[J].电信科学,2010,26(09):84-89
- [7] 朱虎明, 罗政, 猴水平,等. Android 平台上基于卷积神经网络的手写数字识别实验[J]. 计算机教育, 2017(11):159-162.
- [8] Cotterell K, Welch I, Chen A. An Android Security Policy Enforcement Tool[J]. International Journal of Electronics & Telecommunications, 2015, 61(4):311-320.
- [9] Lin D, Offutt J, Ammann P, et al. Mutation operators for testing Android apps[J]. Information & Software Technology, 2016, 81(C):154-168.
- [10] 姜航. java 开发语言基础下手机软件开发技术的应用[J]. 信息技术与信息化, 2017(11).
- [11] 陈林. 基于 Java 语言的手机软件开发技术分析[J]. 电子技术与软件工程, 2015(6):53-54.
- [12] Ali A H, Sagheer A M. Design of an Android Application for Secure Chatting[J]. International Journal of Computer Network & Information Security, 2017, 9(2):29-35.
- [13] 杨杰.基于 Android 平台的校园新闻客户端的设计与研发[J].现代计算机,2013(20):64-68
- [14] 孟娟娜.基于 Android 平台的移动电子商务系统设计与实现[J].电子设计工程,2016,24(08):27-29+33
- [15] 疯狂 Android 讲义（第 3 版），李刚，电子工业出版社，2017-01-01

致谢

原本觉得毕业离我很遥远，但时间真的过得太快了，寒假时候还觉得毕设时间那么长、很充足，但毕业设计到这里就真的结束了，在程序开发的这段时间里，我遇到很多困难，也自暴自弃了一段时间，感谢舍友、同学和老师给予我的帮助和支持，没有你们我的毕设会是一副什么样子还不得而知。

在此，特别感谢我的毕设导师——巩晶老师，虽然巩晶老师在我做毕设的这段时间里事务繁忙，但是仍然经常关心我们的毕设情况，检查进度并给我们意见，让我不至于因为懒惰等原因拖慢进度甚至无法完成任务。

大学四年，如白驹过隙，仿佛昨天还刚和父母走进这个校园，和辅导员邓佳老师有了许多交流，也得到了邓佳老师的许多关照，让我在为人处世方面学到了很多，懂得了该如何跟人相处，该如何利用大学宝贵的时间来武装自己，同时，在大一的时候，由于班主任刘传文老师的尽心尽责，也让我对大学的学习态度有了新的认识，也非常感谢刘传文老师在后面为我提供的帮助和鼓励，让我的留学申请变得更加顺利。

四年光阴，一路走来，感谢你们！感谢班主任林泓老师对班级的照顾，对我们学习的指导！感谢所有的老师，老师你们辛苦了！你们教会了我很多知识，这都是我一生宝贵的财富！感谢所有的同学，让我得到了以前从未有过的体验，四年，能遇到你们这是这辈子的幸运！

回想四年前，刚刚高考完的我，完全不会想到几年后我会是现在这副模样，原本以为只是上所普普通通的大学，普普通通的完成学业，然后就顺势开始工作，进入社会，和父母一样平平淡淡的走下去，但是武汉理工改变了我，在这里，我看到了很多，了解到了很多，学到了更多，我从未想过我身边会出现那么多可以一起开怀大笑的朋友，也没想到会有这么几个可爱可亲的舍友，几年的同居生活，让我们变得就像亲兄弟一样，这是身为独生子女的我从未有过的体验，而接下来，我还会继续迈开脚步向前走，我从未想过，我也会有朝一日走出国门，去外面看看，希望接下来的几年我能在外好好学习，靠自己的努力和毅力为自己谋求更好的生活，收获属于自己的未来。

感谢的话已不需多说，我们一路走来，我们携手并进，我们，永远不说再见！