# 开发技术文档，安卓部分

## 功能说明

安卓部分的功能现如今主要有两大块儿,写的比较挤，都放在了

GraduationProject/app/glaucoma\_perimetry\_system/src/main/java/com/DefaultCompany/glaucoma\_perimetry\_system/UnityPlayerActivity.java

下，你有时间可以拆开来  
1.读取指定目录下的文件，合并文件并为之追加上文字描述

先贴代码

描述一下功能，大致写了下，逻辑不是很严密，有待优化

这个方法对外指暴露了一个参数，resourcePath，unity计算后生成图片结果的位置

安卓为：/data/data/com.DefaultCompany.glaucoma\_perimetry\_system/files

ios为：/Application/包名/Documents

这个路径是为下面BitmapFactory.*decodeFile读取图片准备的*



private void mergeBitmap(String resourcePath) throws IOException {

Bitmap firstBitmap = BitmapFactory.*decodeFile*(resourcePath+"/左眼GrayScale.png");  
 System.*out*.println(firstBitmap);  
 Bitmap secondBitmap = BitmapFactory.*decodeFile*(resourcePath+"/右眼GrayScale.png");  
 System.*out*.println(secondBitmap);  
 Bitmap bitmap;

这里进行了一个简单的判断，如果结果图同时存在，则进行合并，否则的话只拿到存在的一张然后贴上文字输出，但之后如果再添加新的结果图，这段代码就不再适用了，所以你看情况优化  
 if(firstBitmap!=null&&secondBitmap!=null){  
 int w1 = firstBitmap.getWidth();  
 int h1 = firstBitmap.getHeight();  
 int w2 = secondBitmap.getWidth();  
 int h2 = secondBitmap.getHeight();

创建一个bitmap  
 bitmap= Bitmap.*createBitmap*(w1+w2, h1 , firstBitmap.getConfig());  
 Paint paint=new Paint();

设置字体样式  
 paint.setStyle(Paint.Style.*FILL*);  
 paint.setTextSize(100);  
 Canvas canvas = new Canvas(bitmap);  
 canvas.drawRGB(255, 255, 255);

将加载的bitmap涂到canvas上  
 canvas.drawBitmap(firstBitmap, new Matrix(), null);

涂文字  
 canvas.drawText("左眼",w1/3, h1/6\*5, paint);

下面的大致逻辑相同，加载图片，创建画板，将图片写到画板上，填上文字，生成图片，最后会放到缓存里getCacheDir()  
 canvas.drawBitmap(secondBitmap, w1, 0, null);  
 canvas.drawText("右眼", w1+w2/2, h2/6\*5, paint);  
 }  
 else {  
 bitmap=(firstBitmap!=null)?firstBitmap:secondBitmap;  
 int w = bitmap.getWidth();  
 int h = bitmap.getHeight();  
 bitmap= Bitmap.*createBitmap*(w,h , bitmap.getConfig());  
 String eye=(firstBitmap!=null)?"左眼":"右眼";  
 Paint paint=new Paint();  
 paint.setStyle(Paint.Style.*FILL*);  
 paint.setTextSize(100);  
 Canvas canvas = new Canvas(bitmap);  
 canvas.drawRGB(255, 255, 255);  
 canvas.drawBitmap(firstBitmap, new Matrix(), null);  
 canvas.drawText(eye,w/3, h/6\*5, paint);  
 }  
 File path = new File(getCacheDir(),"shareImage.png");  
 OutputStream os = new FileOutputStream(path);  
 bitmap.compress(Bitmap.CompressFormat.*JPEG*, 50, os);  
 os.close();  
}

1. 读取指定目录下的文件，通过其他渠道进行分享

这个方法是不需要参数的，这个msg只是我测试用的，你之后可以把它去掉

public void onPress(String msg) throws IOException {

调用上面的合并文件的方法

mergeBitmap("/data/data/com.DefaultCompany.glaucoma\_perimetry\_system/files");

下面这段就是分享功能的主逻辑，

这句话也可以删掉，作用是控制手机的横屏与竖屏，这点我已经再unity中设置了，所以下面这句代码完全可以删掉  
 setRequestedOrientation(ActivityInfo.*SCREEN\_ORIENTATION\_PORTRAIT*);

没看过里面怎么转的，大概意思就是new了一个intent对象，然后给他设置一下其他参数，然后startActivity，他就携带着图片跳到其他Activity（你分享到的应用）里了  
 Intent shareInt = new Intent(Intent.*ACTION\_SEND*);  
 shareInt.setType("image/\*");  
  
 shareInt.putExtra(Intent.*EXTRA\_SUBJECT*, "选择分享方式");  
 // shareInt.putExtra(Intent.EXTRA\_TEXT, msg); // 要分享的内容  
 File shareFile = new File(getCacheDir(),"shareImage.png");

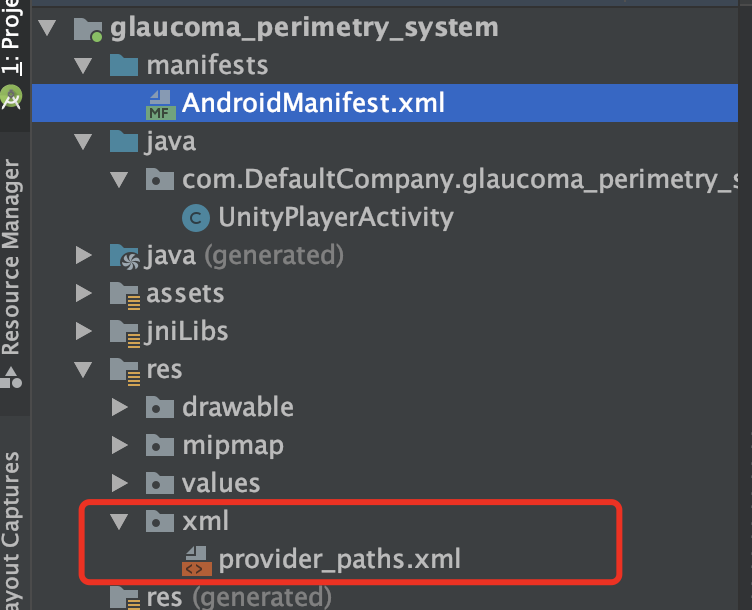
这里安卓必须要把路径进行修饰，安卓自身为了安全考虑，不允许第三方app（分享）直接调用真实路径，会报错的，ios没试过，不清楚

getPackageName()+ ".fileprovider"这个，路径怎么替换的下面说，你也可以去搜一下这个替换逻辑，我看了人家写的挺不错的，但是我给忘了  
 Uri uri= FileProvider.*getUriForFile*(  
 this,UnityPlayerActivity.this.getPackageName()+ ".fileprovider",shareFile  
 );  
 shareInt.putExtra(Intent.*EXTRA\_STREAM*,uri);  
  
 shareInt.setFlags(Intent.*FLAG\_ACTIVITY\_NEW\_TASK*);  
  
  
  
 startActivity(Intent.*createChooser*(shareInt, getTitle()));  
  
}

程序清单

GraduationProject/app/glaucoma\_perimetry\_system/src/main/AndroidManifest.xml

这段配置要放在application里作用就是当里遇到${applicationId}.fileprovider时，及上面的getPackageName()+ ".fileprovider"，他会去@xml/provider\_paths这个文件里加载路径然后做一个伪装伪装逻辑好像是替换了个头



<provider  
 android:name="android.support.v4.content.FileProvider"  
 android:authorities="${applicationId}.fileprovider"  
 android:exported="false"  
 android:grantUriPermissions="true">  
 <meta-data  
 android:name="android.support.FILE\_PROVIDER\_PATHS"  
 android:resource="@xml/provider\_paths" />  
</provider>

provider\_paths.xml

GraduationProject/app/glaucoma\_perimetry\_system/src/main/res/xml/provider\_paths.xml

<?xml version="1.0" encoding="utf-8"?>  
<paths xmlns:android="http://schemas.android.com/apk/res/android">  
 <cache-path name="cache" path="."/>  
 <files-path name="files" path="./"/>  
</paths>

至于按键控制，就更简单了

这个onKeyDown就是监听你按键是否按下，然后之后会传给他一个keyCode代表你按下的那个按键的值，

所以你可以换成蓝牙，无非就是查查看蓝牙对应的KeyEvent是个啥，其他的不用你操心，安卓自己会去解决

@Override public boolean onKeyDown(int keyCode, KeyEvent event) {

这里的逻辑也很简但，判断你按下了那个键，然后调用unity里相应的方法  
 if(KeyEvent.*KEYCODE\_HEADSETHOOK* == keyCode){

这个对应着线控耳机的中键，调用unity里的开始运行程序  
 UnityPlayer.*UnitySendMessage*("LeftEyeButton", "ifStartCheck","");  
 }  
 if(KeyEvent.*ACTION\_UP*==keyCode){

这个对应着线控耳机的前进键，调用unity里的加速运行  
 UnityPlayer.*UnitySendMessage*("Background", "ActionUp","");  
 }  
 return mUnityPlayer.injectEvent(event);  
}