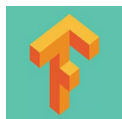


[Android] 随时拍图像处理部分总结及源码分享

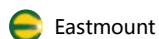
原创 置顶 Eastmount 最后发布于2015-02-07 23:03:39 阅读数 23162 ☆ 收藏

展开



Python+TensorFlow人工智能

该专栏为人工智能入门专栏,采用Python3和TensorFlow实现人工智能相关算法。前期介绍安装流程、基础语法...



¥9.90

去订阅

随着寒假到来,终于有时间总结下Android这个随手拍的课程作业了,同时分享了我完成的这部分对图像处理的心得吧!你可以结合demo来阅读这篇文章,希望对大家有所帮助.

如果你绝对下面文字太过啰嗦,可以直接下载demo.如果你要做android图像处理相关的App,可能对你有些启发.如果文章中有不足之处,请海涵!仅想分享些东西罢了~

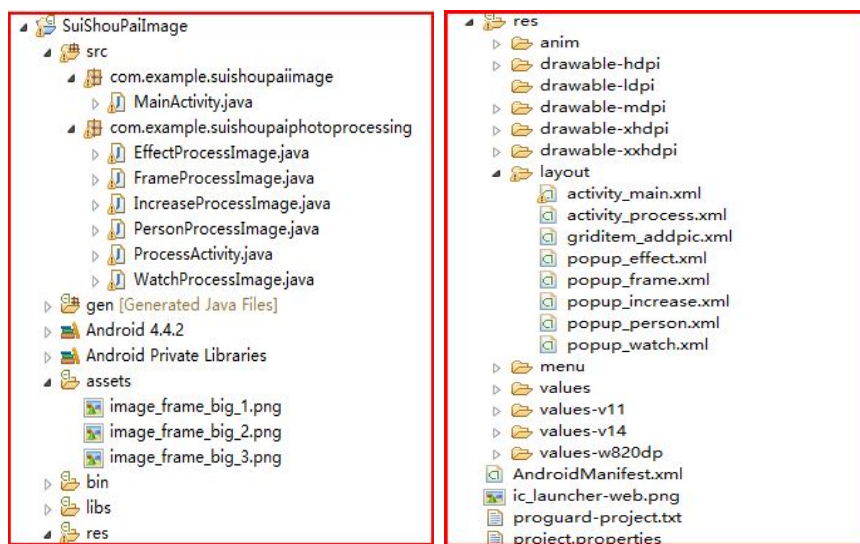
demo下载地址:

<http://download.csdn.net/detail/eastmount/8432127>

<http://pan.baidu.com/s/1c0kz3by>

一. 项目结构及界面

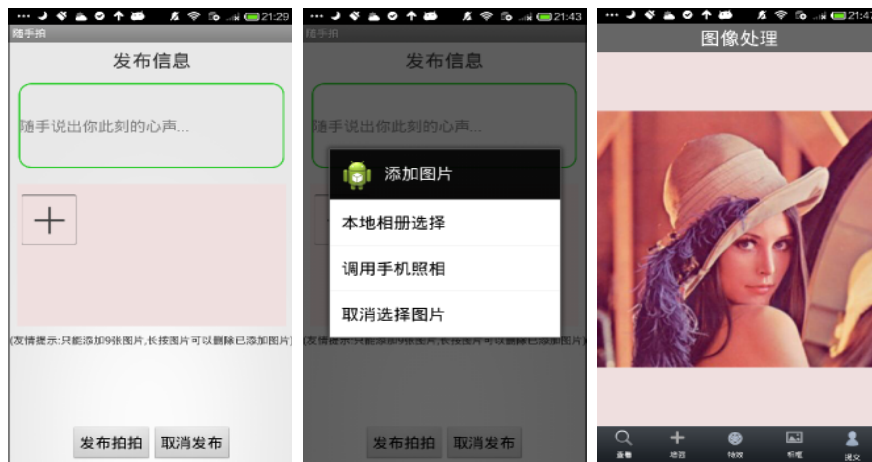
项目工程结构如下图所示:



个人还是比较欣赏这个布局的,其中MainActivity.java是主界面,ProcessActivity.java是图像处理界面,而剩余五个java对应五个不同的图像处理算法: EffectProcessImage(图像特效)、FrameProcessImage(添加相框)、IncreaseProcessImage(图像增强)、PersonProcessImage(图像交互)和WatchProcessImage(查看图片).

同时Layout中有五个xml子布局,采用PopupWindow形式显示五个不同的处理,同时该demo比较好移植,将文件依次复制过去并且MainActivity改成子活动,即可调用.

其中部分界面如下:



MainActivity主界面

添加图片选项

图像处理子界面(底部五个)

二. 图像查看功能

点击主界面GridView中"+"添加图片后,可以选择从相册添加或照相,加载图片进行到处理界面后,点击底部"查看"按钮它会有选中的效果(图标颜色变蓝&背景颜色加深).同时PopupWindow弹出如下界面:



其中点击"旋转"后提交的动态图如下所示:



```
public class WatchProcessImage
{
    private Bitmap mBitmap;
    //构造方法
    public WatchProcessImage(Bitmap bmp)
    {
        mBitmap = bmp;
    }

    /*
     * 2.图片水平翻转 -> 反相
     * flag=0表示第一次翻转 flag=1表示翻转会原图 来回切换
     */
    public Bitmap FlipHorizontalImage(Bitmap bmp, int flag)

    /*
     * 3.图片垂直翻转
     * flag=0表示第一次翻转 flag=1表示翻转会原图 来回切换
     */
    public Bitmap FlipVerticalImage(Bitmap bmp, int flag)

    /*
     * 1.图片旋转 45度一次旋转
     * flag=1表示旋转45度 flag=2表示旋转90度 flag*45为度数
     */
    //旋转图片
    public Bitmap TurnImage(Bitmap bmp, int flag)
}
```

参考以前我的博客:

[\[Android\] 拍照、截图、保存并显示在ImageView控件中](#)

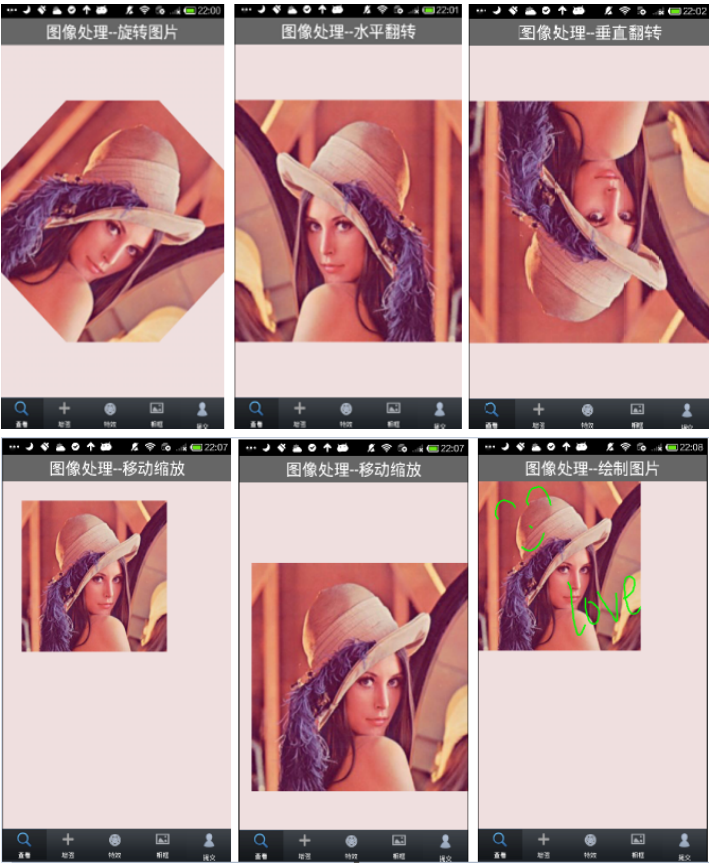
[\[Android\] 使用Matrix矩阵类对图像进行缩放、旋转、对比度、亮度处理](#)

[\[Android\] 触屏setOnTouchListener实现图片缩放、移动、绘制和添加水印](#)

其他效果如下,同时存在几个问题:

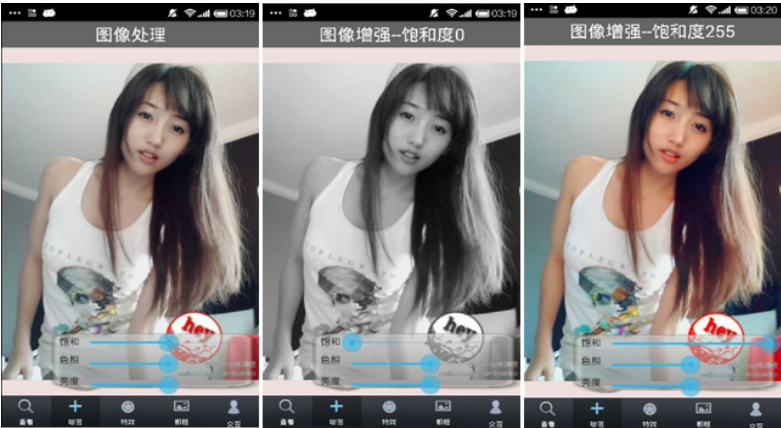
- 1.图像绘图保存我没有做
- 2.移动和缩放图像,上传的都是原图大小

3.添加水印功能感觉简单就没实现



三. 图像增强功能

图像增强主要是通过3个进度条SeekBar实现,可以调节图像的饱和度、亮度和色相.当饱和度等于0时就是黑白二值图像,三个滑动条可以相互调节.如下图所示:





参考我的以前博客：

[\[Android\] 使用Matrix矩阵类对图像进行缩放、旋转、对比度、亮度处理](#)

[\[Android\] 图像处理整合之处处理ColorMatrix和Intend传递路径显示图像](#)

[\[Android\] 底部菜单布局+PopupWindows实现弹出菜单功能\(初级篇\)](#)

同时需要注意一个问题：设置滑动条时需要设置其区间,同时在弹出PopupWindow注意其宽度和不同手机的适屏处理.

该部分算法处理代码如下,每部分算法调用后返回值都是bitmap,然后调用imageShow.setImageBitmap(mbmp);直接显示.

```
//自定义类实现图像增强效果处理 包括:饱和度和、色相、亮度
public class IncreaseProcessImage
{
    private Bitmap mBitmap;

    //构造函数
    public IncreaseProcessImage(Bitmap bmp)
    {
        mBitmap = bmp;
    }

    /*
     * 设置饱和度和值
     */
    public void setSaturation(int value) {}

    /*
     * 设置色相值
     */
    public void SetHue(int value) {}

    /*
     * 设置亮度值
     */
    public void SetLum(int value) {}

    /*
     * 图像增强
     * 饱和度处理 色相处理 亮度处理
     * flag=0表示是否改变饱和度 flag=1表示是否改变色相 flag=2表示是否改变亮度
     */
    public Bitmap IncreaseProcessing(Bitmap bmp, int flag){}
```

四. 图像特效功能

PopupWindow弹出如下界面,主要包括：怀旧、浮雕、光照、素描和锐化处理.



其中特效效果和EffectProcessImage.java如下所示：



```
//自定义实现图像效果处理 包括特效、浮雕、光照、素描、锐化
public class EffectProcessImage
{
    private Bitmap mBitmap;

    //构造方法
    public EffectProcessImage(Bitmap bmp)
    {
        mBitmap = bmp;
    }

    /*
     * 1.图片怀旧处理
     */
    public Bitmap OldRemeberImage(Bitmap bmp) {}

    /*
     * 2.图片浮雕处理
     * 底片效果也非常简单,将当前像素点的RGB值分别与255之差后的值作为当前点的RGB
     * 灰度图像:通常使用的方法是gray=0.3*picR+0.59*picG+0.11*picB
     */
    public Bitmap ReliefImage(Bitmap bmp) {}

    /*
     * 3.图片光照效果
     */
    public Bitmap SunshineImage(Bitmap bmp) {}
}
```

参考我的博客:

[\[Android\] 通过Menu实现图片怀旧、浮雕、模糊、光照和素描效果](#)

[\[Android\] 图像各种处理系列文章合集 \(草稿 强推\)](#)

同时算法中存在一些问题,一个就是效率不高,有好友建议C写算法来调用,但是该文章仅是课堂作业,提供一种方法供大家参考.同时在图像素描算法效果不好,锐化处理应该也不对(可参考博客),光照效果是从中心向边沿映射,用户可自己设置光照直径和方向.



五. 图像相框添加

最后是图像相框合成,如下图所示.我采用的是照相后合成图片:



```
public class FrameProcessImage
{
    private Bitmap mBitmap;
    //构造方法
    public FrameProcessImage(Bitmap bmp)
    {
        mBitmap = bmp;
    }

    /*
     * 3.图片合成 模式三 模式二 模式一
     * 载入相框不同 显示效果不同 方法都是一样的
     */
    public Bitmap addFrameToImage(Bitmap bmp, Bitmap frameBitmap)
    {
        bmp原图 frameBitmap资源图片(边框)
    }

    /*
     * 4.圆角矩形图片相框
     */
    public Bitmap RoundedCornerBitmap(Bitmap bitmap) {}

    /*
     * 5.原型图像相框
     */
    public Bitmap RoundedBitmap(Bitmap bitmap){}

    /*
     * End
     */
}
```

添加相框功能主要是通过两张图片的合成,我也见到过四张图片分别合成相框的效果.但我更喜欢这种两张图片合成的,在assets文件夹中加载透明图片.详见:

[\[Android\] 给图像添加相框、圆形圆角显示图片、图像合成知识](#)

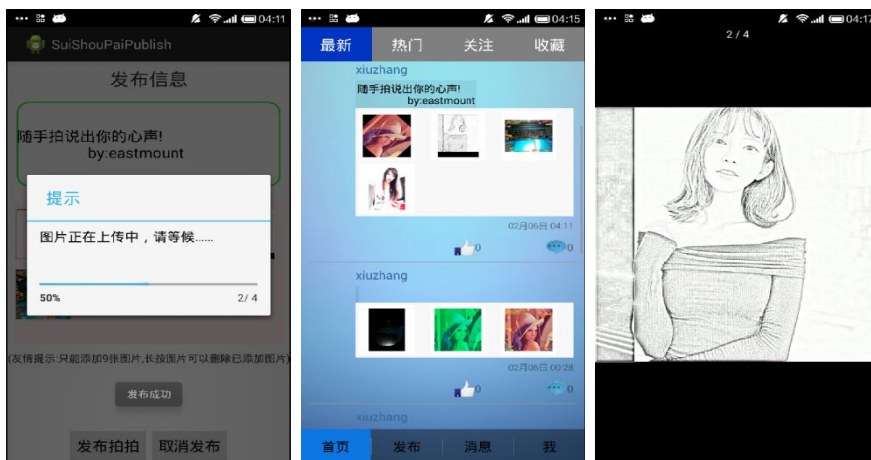
六. 总结及感想

该处理过程中的底部五个按钮切换背景和颜色详见代码ProcessActivity.java, 主要是通切换两张图片和背景实现.如果每个界面显示不同内容,建议使用Fragment滑动实现.参考:

[\[Android\] 使用Include布局+Fragment滑动切换屏幕](#)

[\[Android\] 通过GridView仿微信动态添加本地图片](#)

随手拍我还是比较满意的,其中后台数据库我们使用的是新浪SAE做的,同时ListView刷新拍拍也非常不错!效果如下:



上传的demo我把发布功能删除了,它是通过线程和Process进度条来上传多张图片的.同时ListView可以刷新最近发布消息,点击图片可以查看大图.可以对发布的拍拍进行评论、关注、点赞等操作.

但是由于整个项目是大家一起完成,包括网络、数据库(SAE)、ListView(引用开源),所以只能共享我做的那部分.抱歉~如果有时间可以写些关于Android网络方面的文章,包括如何上传图片(URL链接存储在云盘中)、获取数据库内容等操作.

PS: 最近其实挺烦躁的,一方面这学期确实忙成狗了,寒假也还有个作业.但在火车上我还是挺平静的——听着调皮的钢琴曲,看着窗外流逝的繁华,不知道从什么时候自己已经喜欢上了这种26小时独处的感觉.感受颇多啊!另一个挺令我心烦的就是回家后不久就去做了个手指的小手术,今天写完这篇文章,心情稍微好点!蓦然回首,突然发现这辈子我最对不住的人居然是我自己,希望来年对自己有点,尤其是对自己的身体.同时有机会,找份实习吧!但幸运的是,每每分享一些博客、完成一些项目后都能给我带来一份快乐.

希望文章对大家有所帮助~

最后用最近看的《老人与海》结束这篇文章：

生活总是让我们遍体鳞伤,但到后来,那些受伤的地方一定会变成我们最强壮的地方.

(By:Eastmount 2015-2-7 夜11点 <http://blog.csdn.net/eastmount/>)

👍 点赞 13 ☆ 收藏 ➦ 分享 ...



Eastmount 博客专家

发布了445 篇原创文章 · 获赞 5981 · 访问量 487万+

他的留言板

关注