

Android 示例程序 使用说明书

本手册中所提及的其它软硬件产品的商标与名称，都属于相应公司所有。

本手册的版权属于中国大恒（集团）有限公司北京图像视觉技术分公司所有。未得到本公司的正式许可，任何组织或个人均不得以任何手段和形式对本手册内容进行复制或传播。

本手册的内容若有任何修改，恕不另行通知。

© 2019 中国大恒（集团）有限公司北京图像视觉技术分公司版权所有

网 站：<http://www.daheng-imaging.com>

销售信箱：sales@daheng-imaging.com

销售热线：010-82828878 转 8068

支持信箱：support@daheng-imaging.com

支持热线：400-999-7595

目录

- 1. 示例程序功能说明 1
 - 1.1. Apk 安装说明 1
 - 1.2. 主界面..... 2
 - 1.3. 功能属性界面 2
 - 1.3.1. 触发模式设置 2
 - 1.3.2. 曝光增益设置 3
 - 1.3.3. 白平衡设置 3
 - 1.3.4. 图像处理设置 4
 - 1.3.5. 用户参数组控制..... 4
 - 1.3.6. 版本信息..... 4
- 2. 三种采集方式切换 6
 - 2.1. 采集控制 6
 - 2.2. 不同采集接口的使用 6
- 3. 版本历史 7

1. 示例程序功能说明

在 Android SDK 包中的 app 目录下有三个版本的 Apk 安装包，分别是 GxViewer_GetBitmap、GxViewer_GetRawImage、GxViewer_GetImageBySurface。三个版本的显示界面和功能基本一致，不同点在于：

- 1) 使用的采集接口不同；
- 2) 保存的图像格式不同：GxViewer_GetBitmap、GxViewer_GetRawImage 保存的图像格式为 jpeg 格式，GxViewer_GetImageBySurface 保存的图像为 raw 格式。

示例程序源码位于 sample 目录下。

1.1. Apk 安装说明

- 1) 将 Apk 安装文件拷贝至 Android 设备中；
- 2) 在 Android 设备中的文件管理器中找到 Apk 安装文件；
- 3) 根据提示正确安装 Apk 安装包；
- 4) 安装完成后，启动程序，将会出现请求文件读写权限的提示框，如下图所示，请选择“始终允许”，否则无法保存图像；



- 5) 将 U3V 相机连接到 Android 设备上后，会弹出请求设备权限提示框，如下图所示，请点击“确定”。



1.2. 主界面

相机正常采图后的界面如下图所示，主界面主要由三部分构成，图像显示区、信息显示区和保存图像按钮。



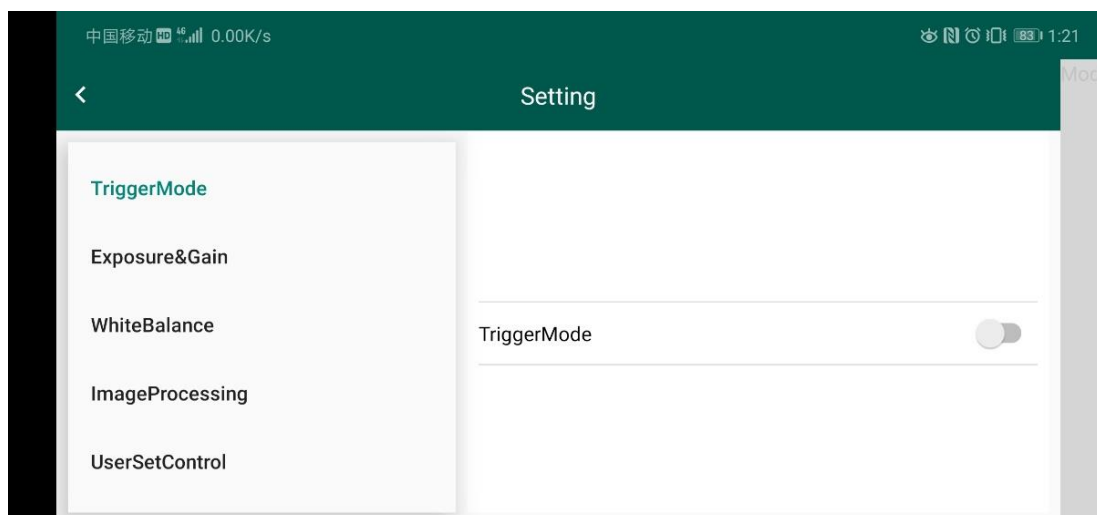
- 1) 图像显示区：显示采集到的图像；
- 2) 保存图像按钮：点击该按钮，在连续采集模式下将会保存当前所采集的图像，在触发模式下将会首先发送软触发信号，随后保存采集到的图像。图像保存在 Android 设备的 Pictures 目录下，保存成功后可在图库中查看；
- 3) 信息显示区：在主界面的左上角依次显示相机型号、采集帧数、采集帧率。

1.3. 功能属性界面

在主界面中向右滑动可呼出设置界面，设置页面中包括：触发模式、曝光增益、白平衡、图像处理、用户参数组控制、版本信息。

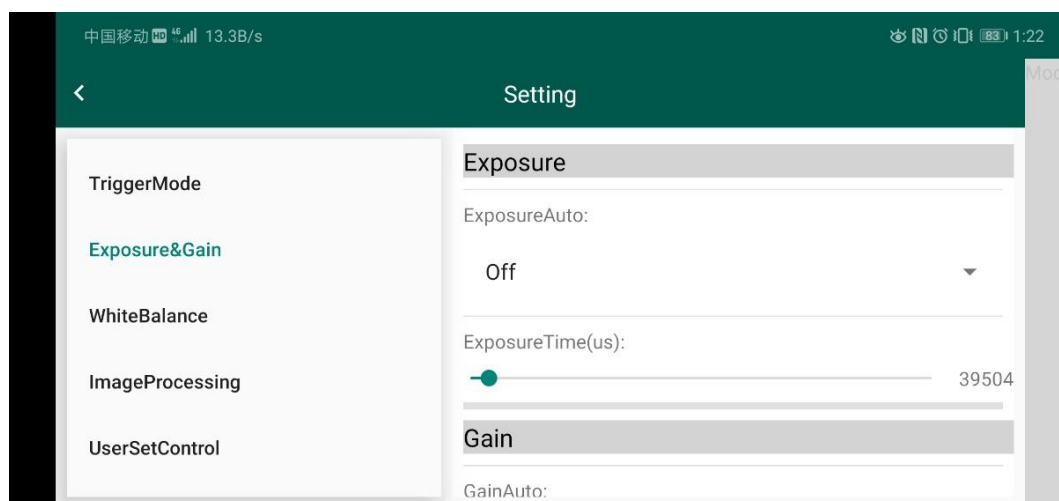
1.3.1. 触发模式设置

触发模式设置如下图所示：

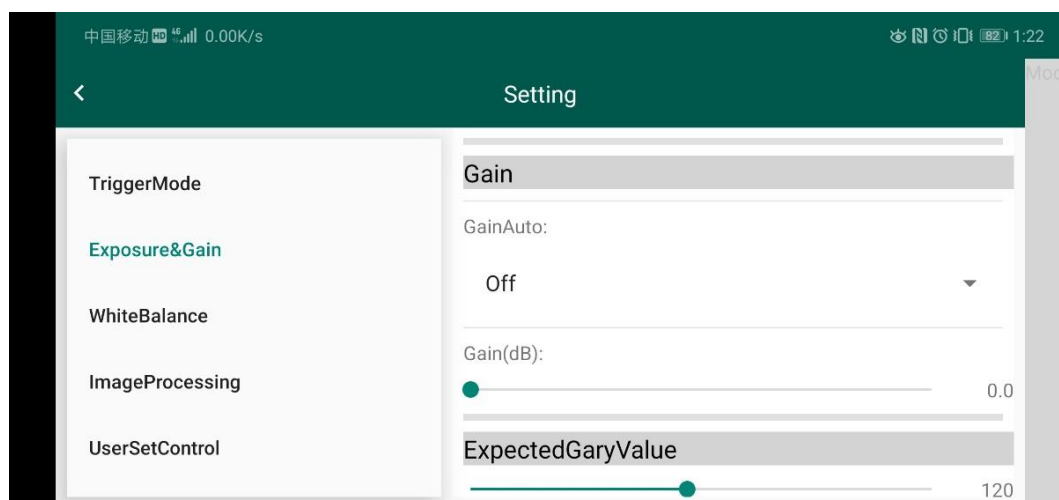


1.3.2. 曝光增益设置

曝光设置如下图所示：

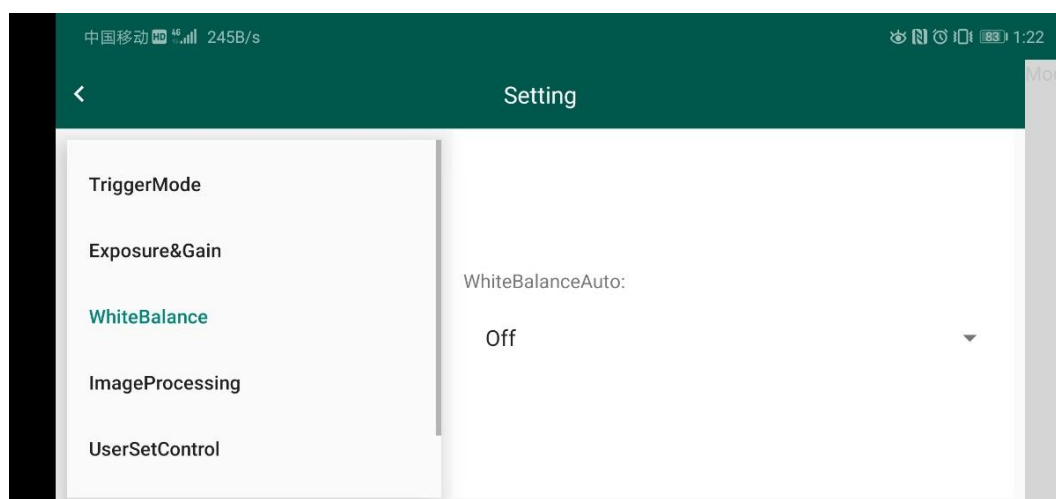


增益、期望灰度值设置如下图所示：



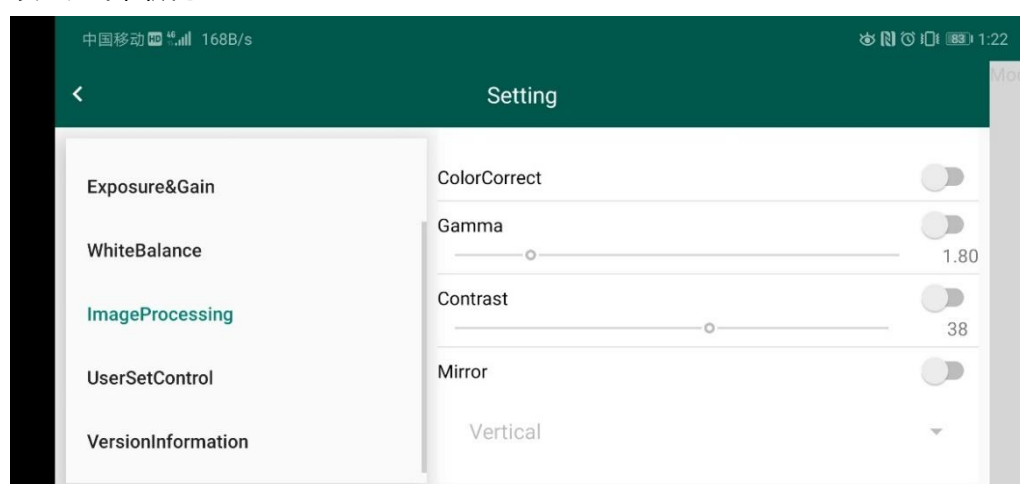
1.3.3. 白平衡设置

白平衡设置如下图所示：



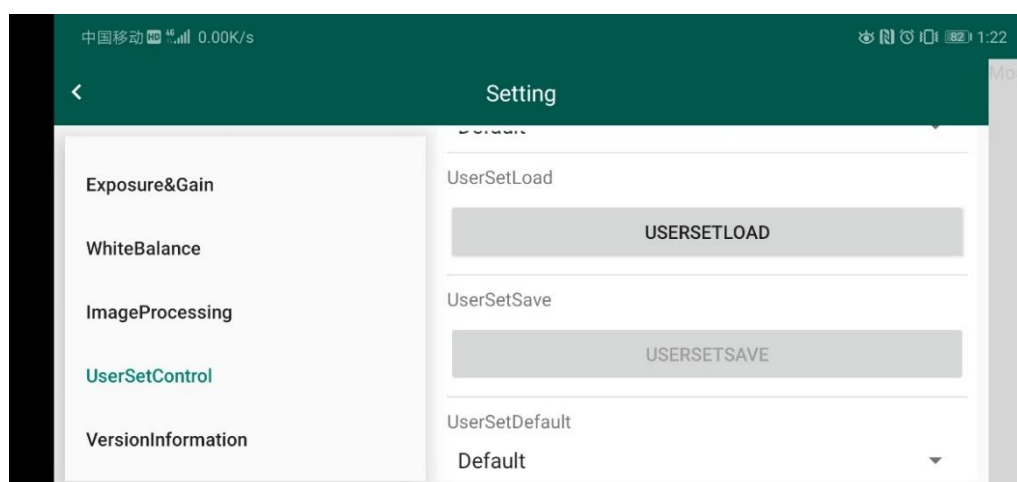
1.3.4. 图像处理设置

图像处理设置如下图所示：



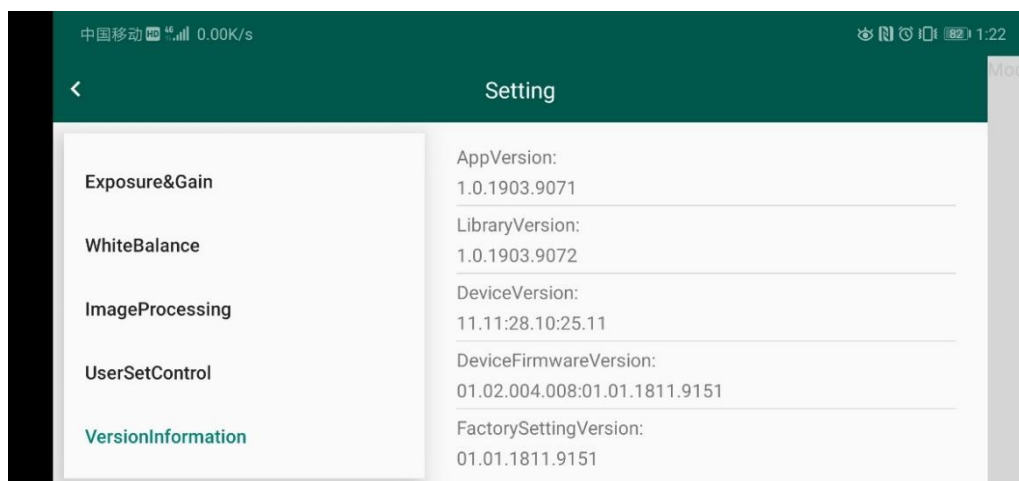
1.3.5. 用户参数组控制

用户参数组控制如下图所示：



1.3.6. 版本信息

版本信息如下图所示：



2. 三种采集方式切换

示例程序源码工程中包含了三种采集方式的实现，可通过注释的方式切换采集方式。

在 MainActivity.java 中可以找到 startAcquisitionImage 方法，其代码如下所示：

```
void startAcquisitionImage() {
    m_threadShareObj.m_acquisitionImage = new AcquisitionByGetBitmap(m_device, m_threadShareObj);
    // m_threadShareObj.m_acquisitionImage = new AcquisitionByGetImageBySurface(m_device, m_threadShareObj);
    // m_threadShareObj.m_acquisitionImage = new AcquisitionByGetRawImage(m_device, m_threadShareObj);

    // start acquisition image thread
    m_threadShareObj.m_acquisitionImage.start();
}
```

1) 第一行实现启动 getBitmap 的采集过程，取消此行注释，注释另外两行，就切换成 getBitmap 采集方式；（如上图所示）

2) 第二行实现启动 getImageBySurface 的采集过程，取消此行注释，注释另外两行，就切换成 getImageBySurface 采集方式；

3) 第三行实现启动 getRawImage 的采集过程 取消此行注释 注释另外两行 就切换成 getRawImage 采集方式。

注意：在同一个程序中仅可以使用三者之一，否则将会导致不可预知的错误。

2.1. 采集控制

三种采集方式的流程是通过三个继承于 AcquisitionImage 类的子类分别控制的，具体如下：

1) AcquisitionByGetImageBySurface：实现 getImageBySurface 接口采集控制，主要实现采集线程的创建和启停工作；

2) AcquisitionByGetBitmap：实现 getBitmap 接口采集控制，主要实现采集线程和显示线程等 2 个线程的创建和启停工作；

3) AcquisitionByGetRawImage：实现 getRawImage 接口采集控制，主要实现采集线程、图像处理线程和显示线程等 3 个线程的创建和启停工作。

2.2. 不同采集接口的使用

采集过程是在独立的线程中进行的，AcquireByBitmapRunnable、AcquireBySurfaceRunnable、AcquireByRawImageRunnable 等 3 个类分别实现了三种采集接口的采图：

1) AcquireBySurfaceRunnable 实现 getImageBySurface 接口的采图；

2) AcquireByBitmapRunnable 实现 getBitmap 接口的采图；

3) AcquireByRawImageRunnable 实现 getRawImage 接口的采图。

三种接口的详细使用说明见《Java 接口开发说明书》。

3. 版本历史

| 序号 | 修订版本号 | 所做改动 | 发布日期 |
|----|--------|------|------------|
| 1 | V1.0.0 | 初始发布 | 2019-05-08 |