China Daheng Group, Inc. Beijing Image Vision Technology Branch

Android Sample Program

User Manual

Version: V1.0.0

Date: 2019-05-15

DAHENG IMAGING 大恒图像

Web: http://www.daheng-imaging.com
Sales Email: isales@daheng-imaging.com
Sales Tel: +86 10 8282 8878
Support Email: isupport@daheng-imaging.com

Contents

1. Sample Program Function Description

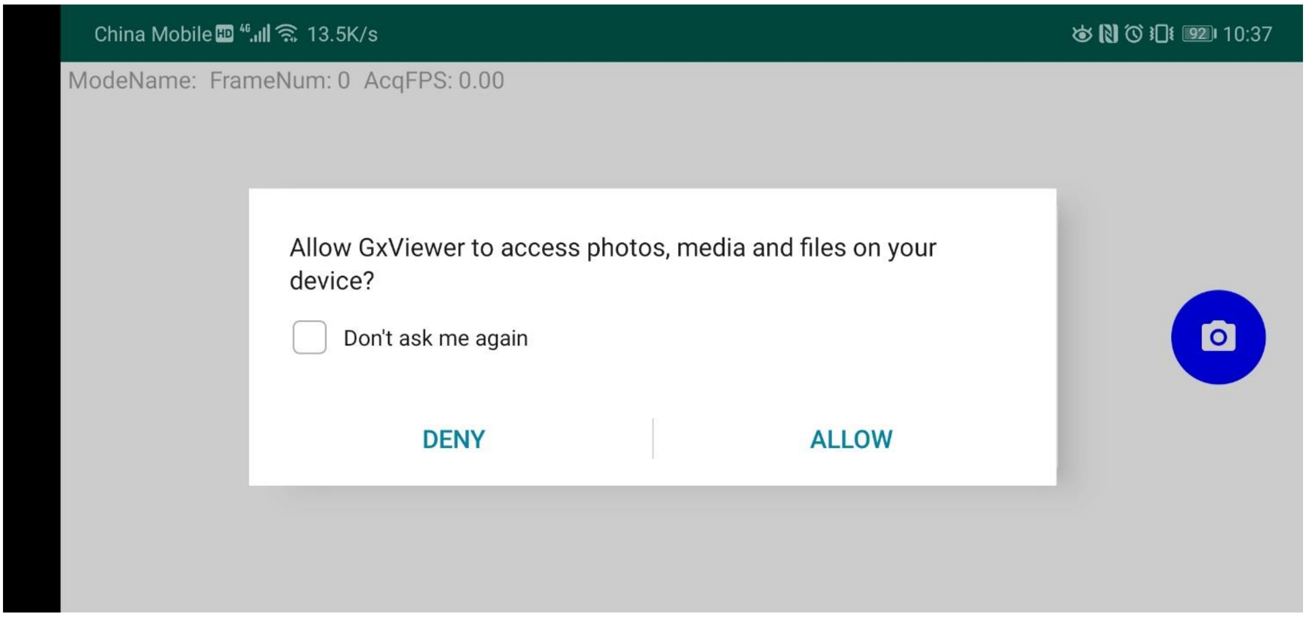
There are three versions of the Apk installation package in the app directory of the Android SDK package, namely GxViewer_GetBitmap, GxViewer_GetRawImage and GxViewer_GetImageBySurface. The display interfaces and functions of the three versions are basically the same, the differences are:

1) The acquisition interfaces are different.
2) The saved image formats are different: GxViewer_GetBitmap and GxViewer_GetRawImage save the image format in jpeg format, and GxViewer_GetImageBySurface saves the image in raw format.

The source code of the sample program is located in the sample directory.

1.1. Apk installation instructions

1) Copy the Apk installation file to your Android device.
2) Find the Apk installation file in the file manager on the Android device.
3) Install the Apk installation package correctly according to the prompts.
4) After the installation is complete, start the program, and a prompt box for requesting read and write permissions of the file will appear. As shown in the figure below, please select "ALLOW", otherwise the image cannot be saved.



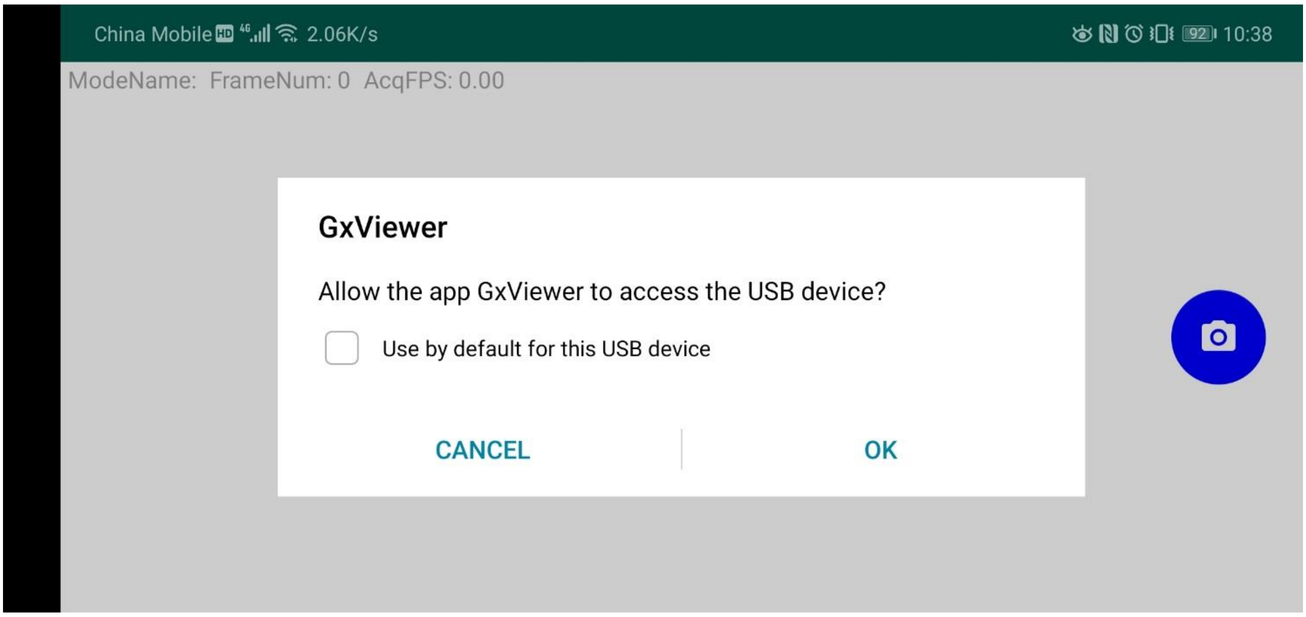
China Mobile 13.5K/s 10:37

ModeName: FrameNum: 0 AcqFPS: 0.00

Allow GxViewer to access photos, media and files on your device?

□ Don't ask me again

5) After connecting the U3V camera to the Android device, a prompt box for requesting device permissions will pop up, as shown in the figure below, please click "OK".



China Mobile 2.06K/s 10:38

ModeName: FrameNum: 0 AcqFPS: 0.00

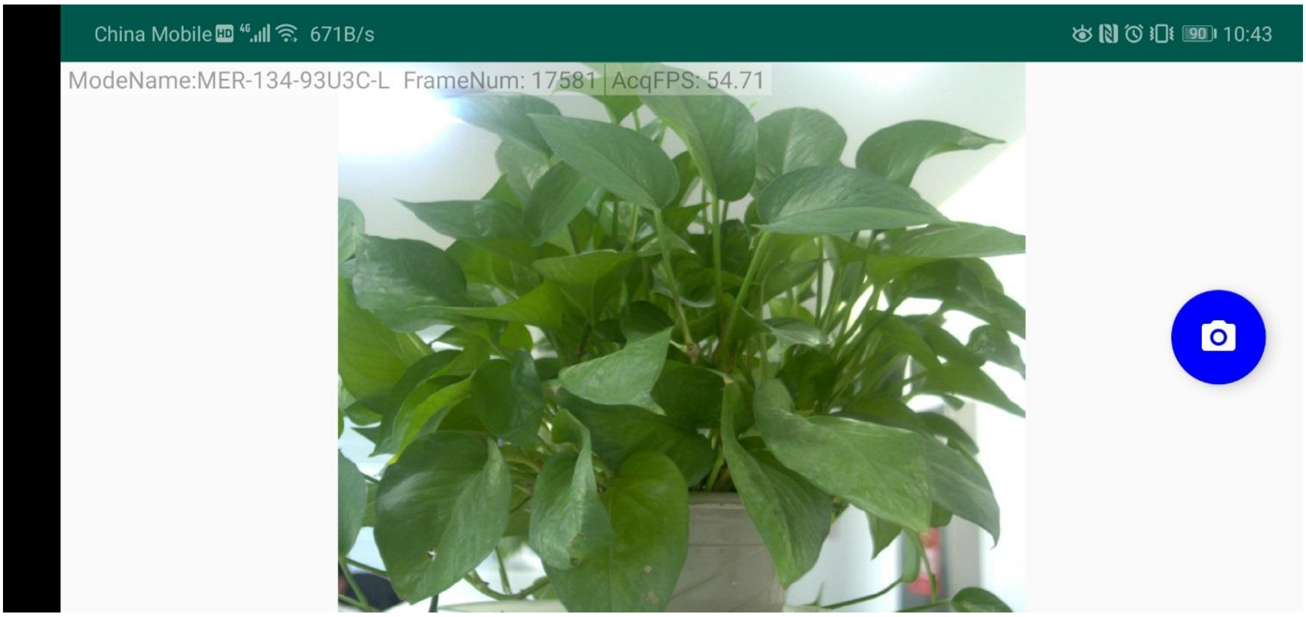
GxViewer

Allow the app GxViewer to access the USB device?

□ Use by default for this USB device

1.2. Main interface

After the camera acquires the image normally, the interface is as shown in the figure below. The main interface consists of three parts: the image display area, the information display area and the save image button.



China Mobile 671B/s 10:43

ModeName:MER-134-93U3C-L FrameNum: 17581 AcqFPS: 54.71

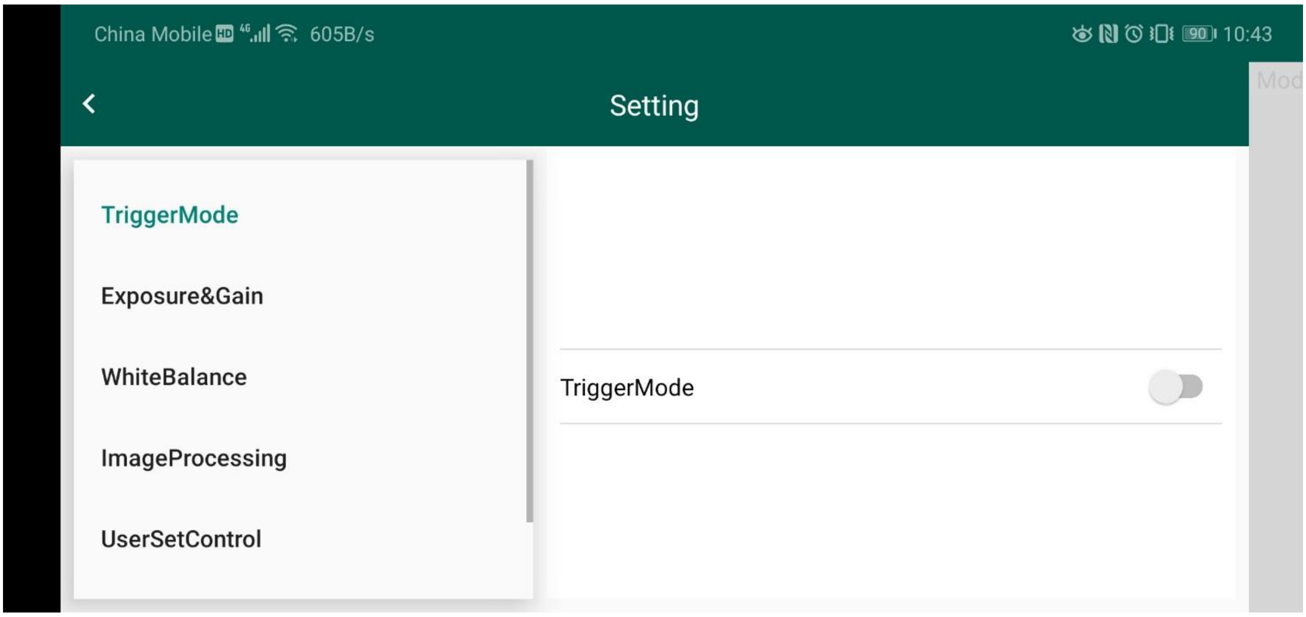
1) Image display area: display the acquired image.
2) Save image button: click this button to save the currently acquired image in continuous acquisition mode. In the trigger mode, the software trigger signal will be sent first, and then the acquired image will be saved. The image is saved in the Pictures directory of the Android device. Once saved successfully, it can be viewed in the gallery.
3) Information display area: display the camera model, the number of acquisition frames, and the acquisition frame rate in the upper left corner of the main interface.

1.3. Function feature interface

Swipe right in the main interface to see the setting interface. The setting interface includes: Trigger Mode, Exposure&Gain, White Balance, Image Processing, UserSet Control, and Version Information.

1.3.1. Trigger Mode setting

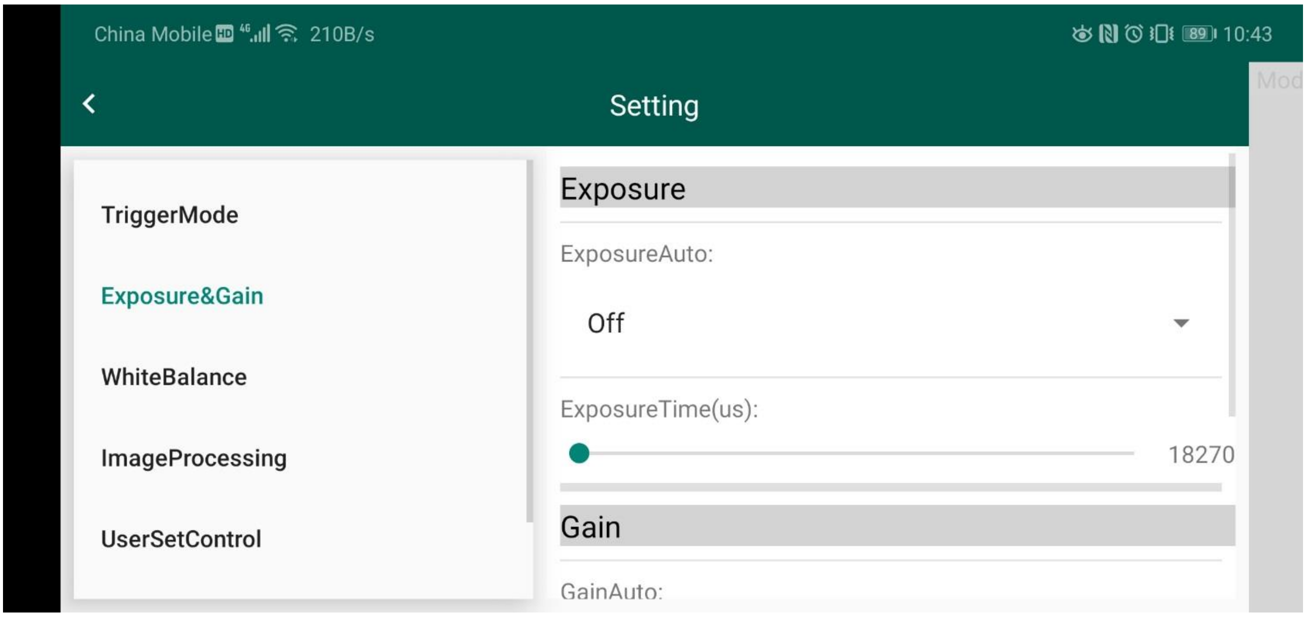
The setting for Trigger Mode is shown below:



TriggerMode

1.3.2. Exposure&Gain setting

The setting for Exposure is shown below:



Exposure

ExposureAuto:

Off

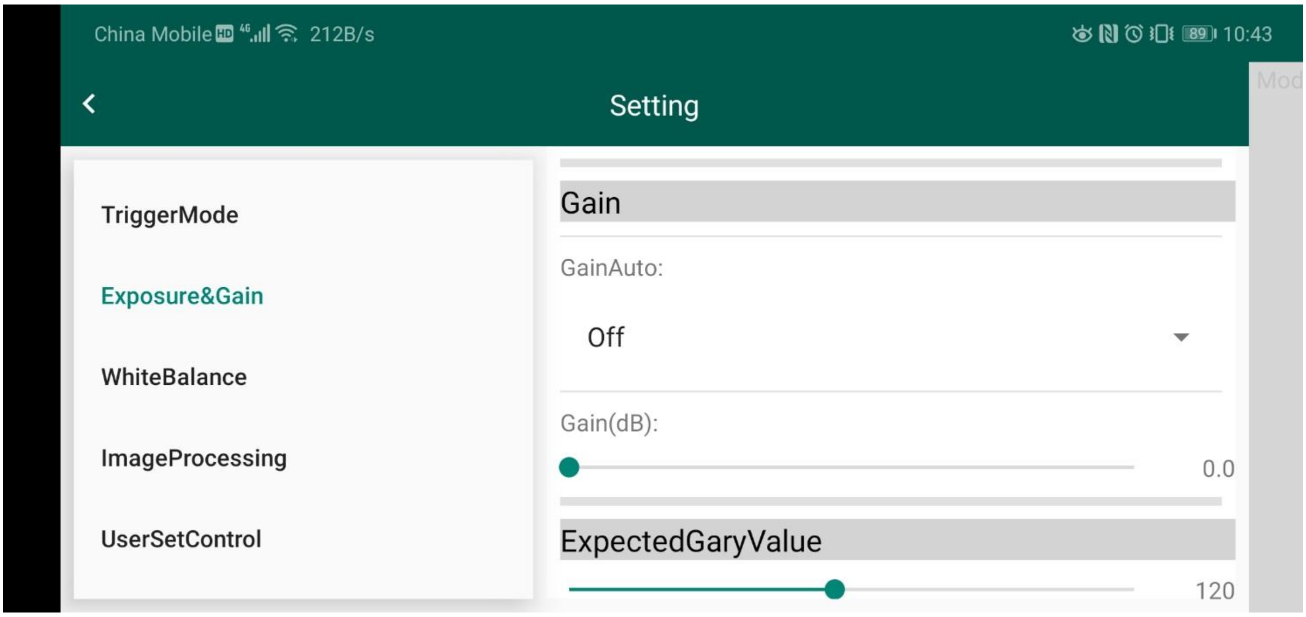
ExposureTime(us):

18270

Gain

GainAuto:

The setting for Gain and Expected Gray Value is shown below:



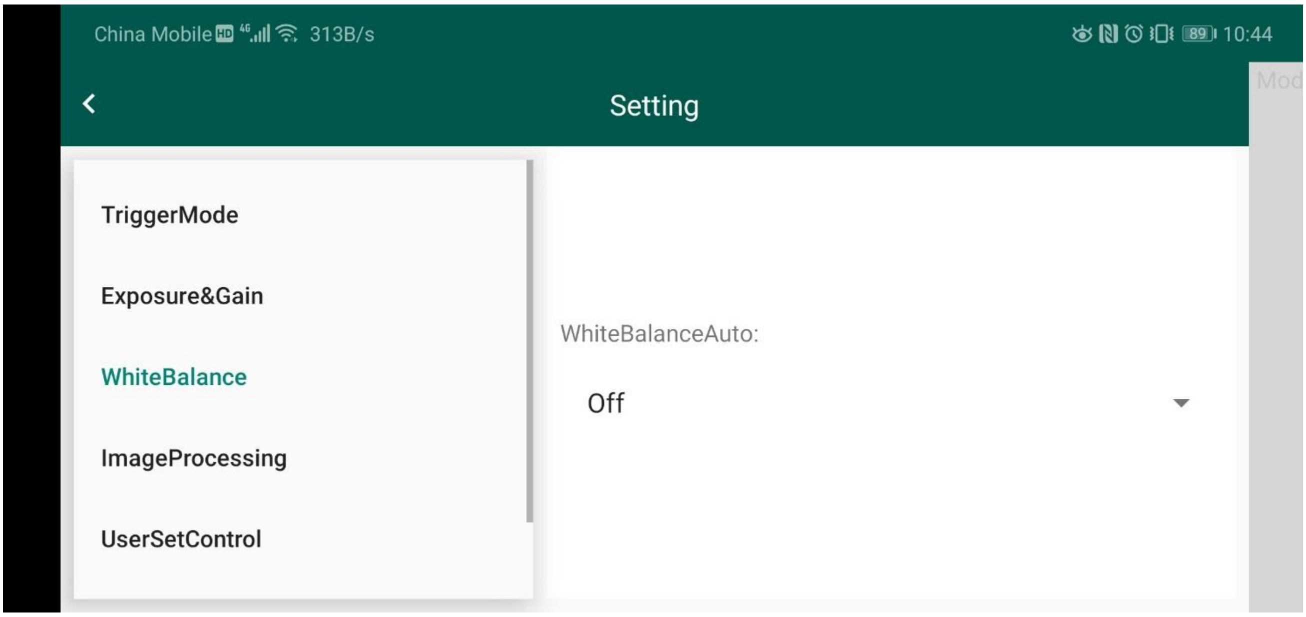
GainAuto: Off

Gain(dB): 0.0

ExpectedGaryValue 120

1.3.3. White Balance setting

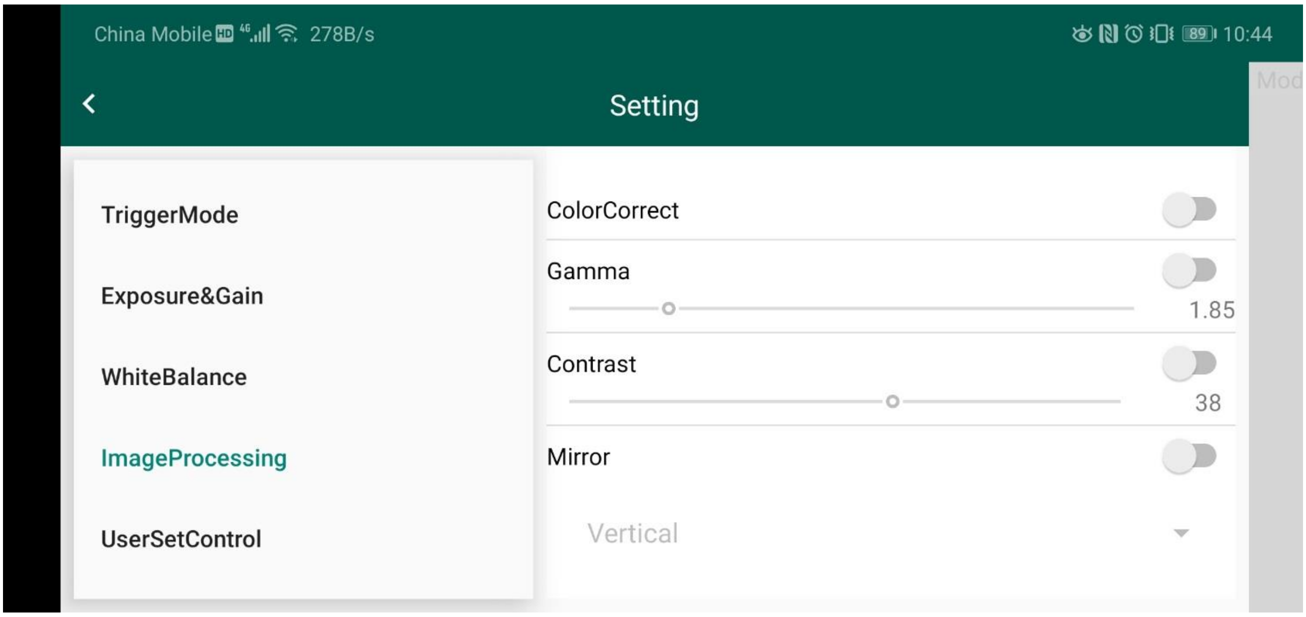
The setting for White Balance is shown below:



WhiteBalanceAuto: Off

1.3.4. Image Processing setting

The setting for Image Processing is shown below:



ColorCorrect

Gamma 1.85

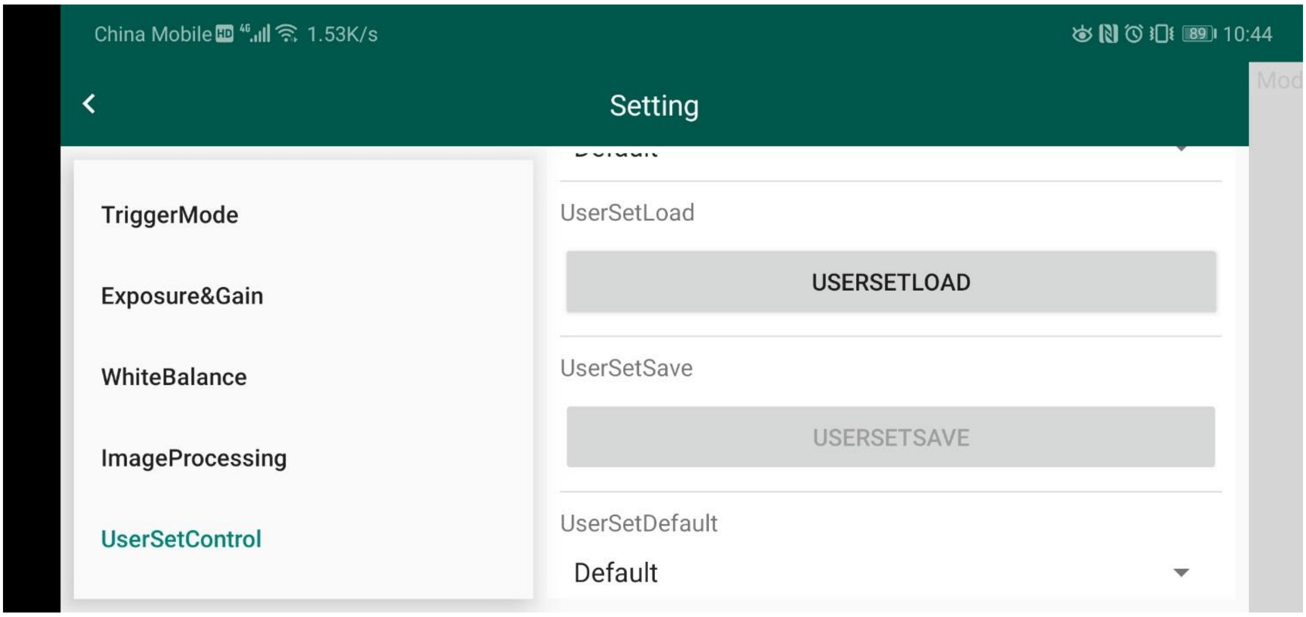
Contrast 38

Mirror

Vertical

1.3.5. UserSet Control

UserSet Control is shown below:



UserSetLoad

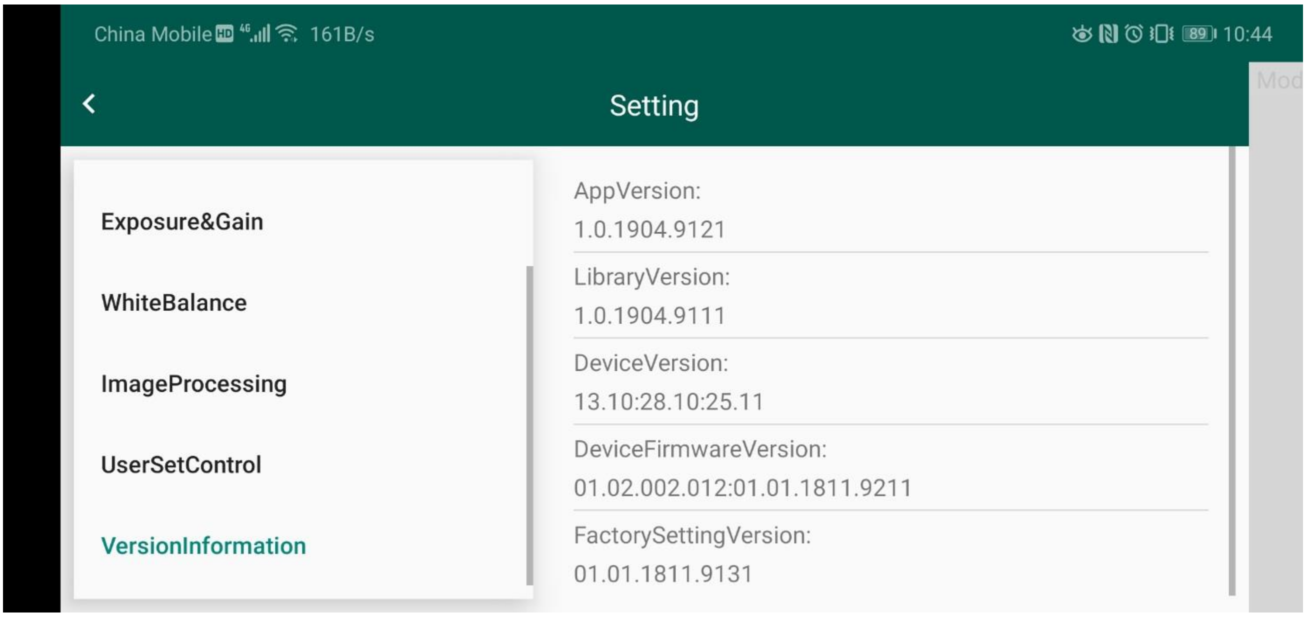
UserSetSave

UserSetDefault

Default

1.3.6. Version Information

Version Information is shown below:



AppVersion:
1.0.1904.9121

LibraryVersion:
1.0.1904.9111

DeviceVersion:
13.10:28.10:25.11

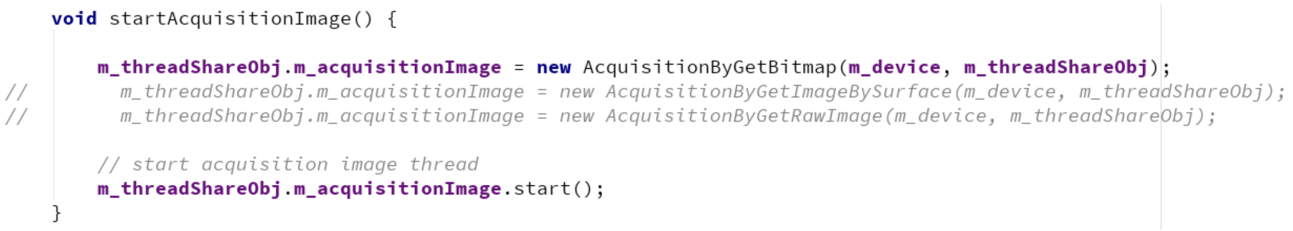
DeviceFirmwareVersion:
01.02.002.012:01.01.1811.9211

FactorySettingVersion:
01.01.1811.9131

2. Switching of Three Acquisition Methods

The sample program source project can implement three acquisition methods, and the acquisition methods can be switched by means of annotation.

The startAcquisitionImage method can be found in MainActivity.java with the following code:



```
    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) The first line implements the start of the acquisition process of getBitmap. Cancel the annotation of this line, and annotate the other two lines, then you can switch to the getBitmap acquisition method (as shown above).
2) The second line implements the start of the acquisition process of getImageBySurface. Cancel the annotation of this line, and annotate the other two lines, then you can switch to the getImageBySurface acquisition method.
3) The third line implements the start of the acquisition process of getRawImage. Cancel the annotation of this line, and annotate the other two lines, then you can switch to the getRawImage acquisition method.

Note: Only one of the three acquisition methods can be used in one program, otherwise it will lead to unpredictable errors.

2.1. Acquisition control

The flow of the three acquisition methods is controlled by three subclasses that inherit from the AcquisitionImage class, as follows:

1) AcquisitionByGetImageBySurface: implement the acquisition control of the getImageBySurface interface, mainly to implement the creation and start or stop of the acquisition thread.
2) AcquisitionByGetBitmap: implement the acquisition control of the getBitmap interface, mainly to implement the creation and start or stop of the acquisition thread and the display thread.
3) AcquisitionByGetRawImage: implement the acquisition control of the getRawImage interface, mainly to implement the creation and start or stop of the acquisition thread, the image processing thread and the display thread.

2.2. Use of different acquisition interfaces

The acquisition process is performed in separate threads. The AcquireByBitmapRunnable, AcquireBySurfaceRunnable, and AcquireByRawImageRunnable classes respectively implement the acquisition of three acquisition interfaces:

1) AcquireBySurfaceRunnable class implements the acquisition of the getImageBySurface interface.
2) AcquireByBitmapRunnable class implements the acquisition of the getBitmap interface.
3) AcquireByRawImageRunnable class implements the acquisition of the getRawImage interface.

Detailed instructions for using the three interfaces can be found in the Java Interface Development User Manual.

3. Revision History

No.	Version	Changes	Date
1	V1.0.0	Initial release	2019-05-15