

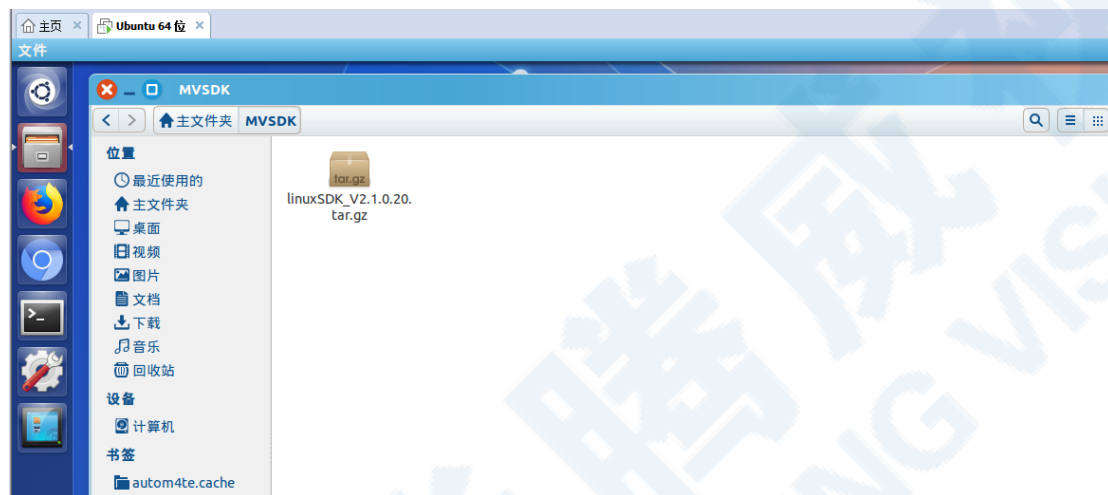
Linux SDK use

1. linux SDK

Technical Support Email: support@huatengvision.com

Boot Linux system, Create folder MVSDK, Copy linuxSDK package to the directory

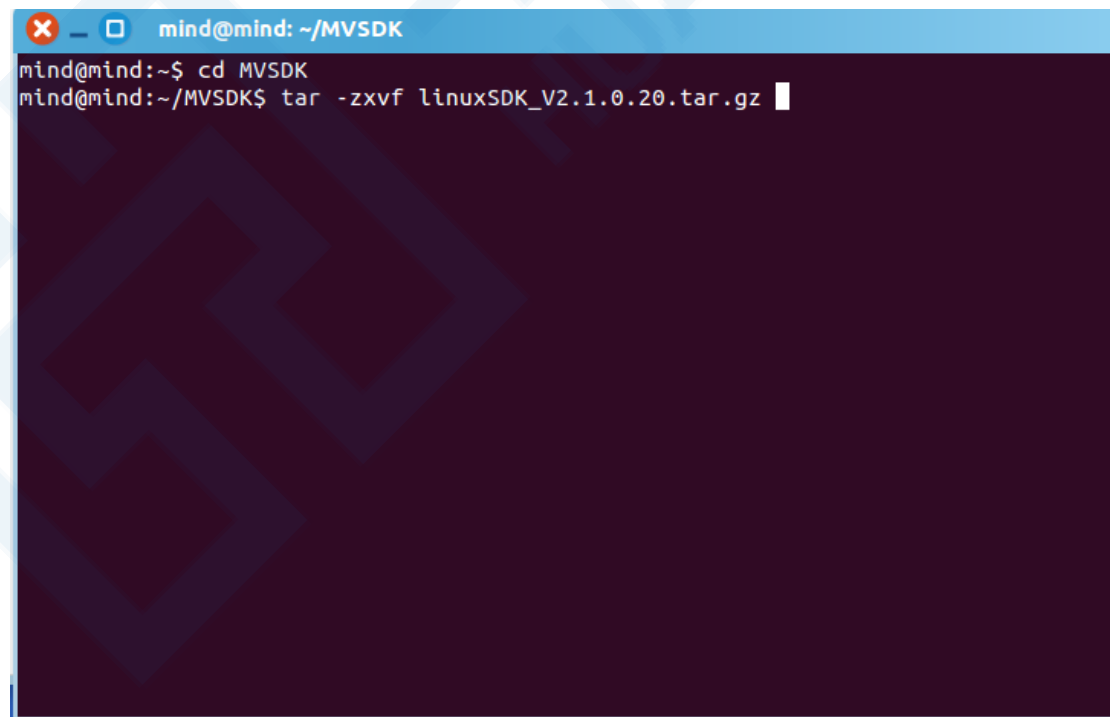
Note: Administrator status is required to run the program.

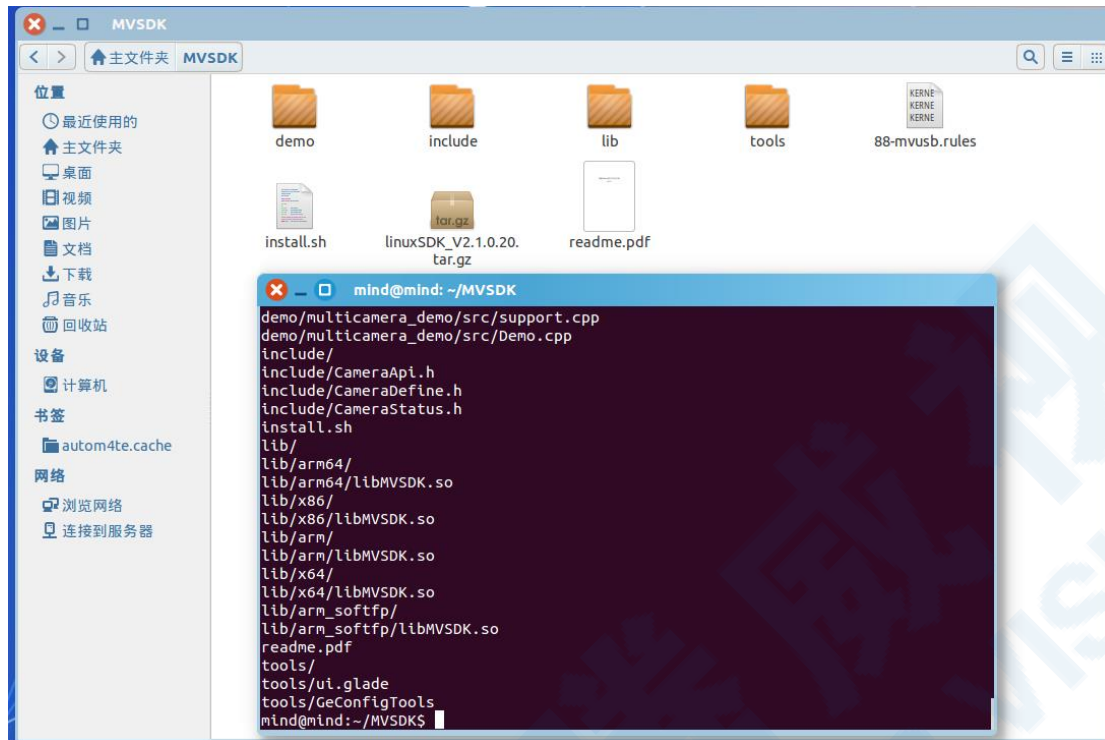


Open terminal

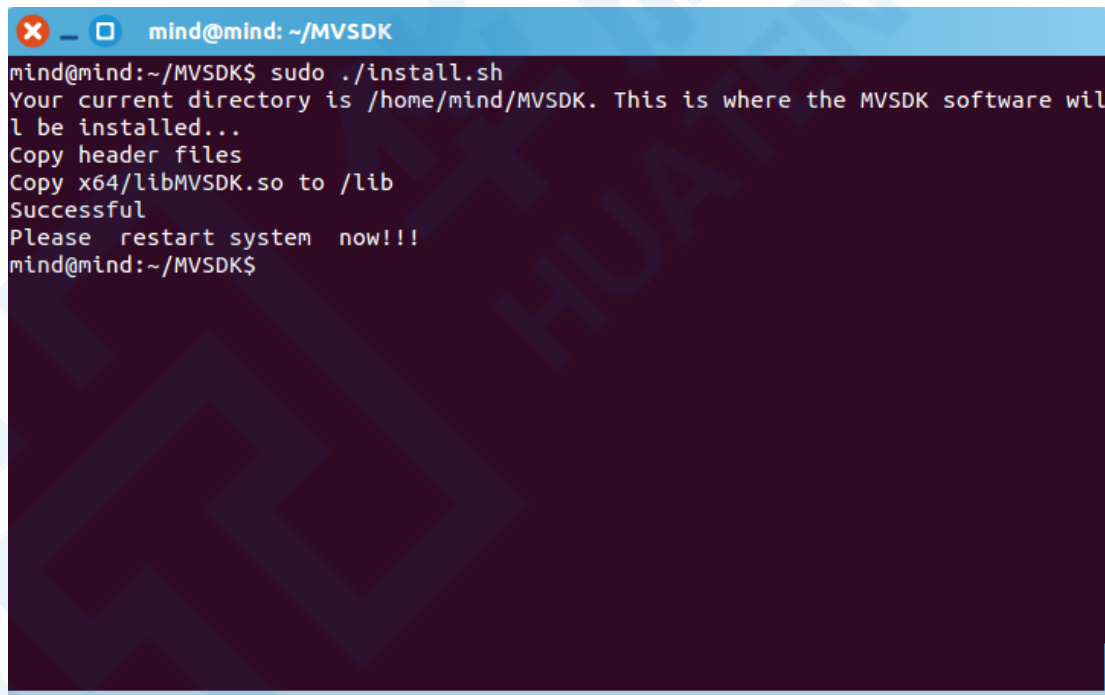
Enter **cd MVSDK** to enter the MVSDK directory

tar -zxvf linuxSDK_V2.1.0.xx.tar.gz Unzip to the current directory





Execute the script, copy the library and header files to the system directory
sudo ./install.sh



If the installation script fails, you can manually copy the library and header files to complete the installation

#copy header file

sudo cp **MVSDK/include/*** /usr/include/

#Copy the corresponding platform library

x86 64-bit system: sudo cp MVSDK/lib/x64/libMVSDK.so /lib

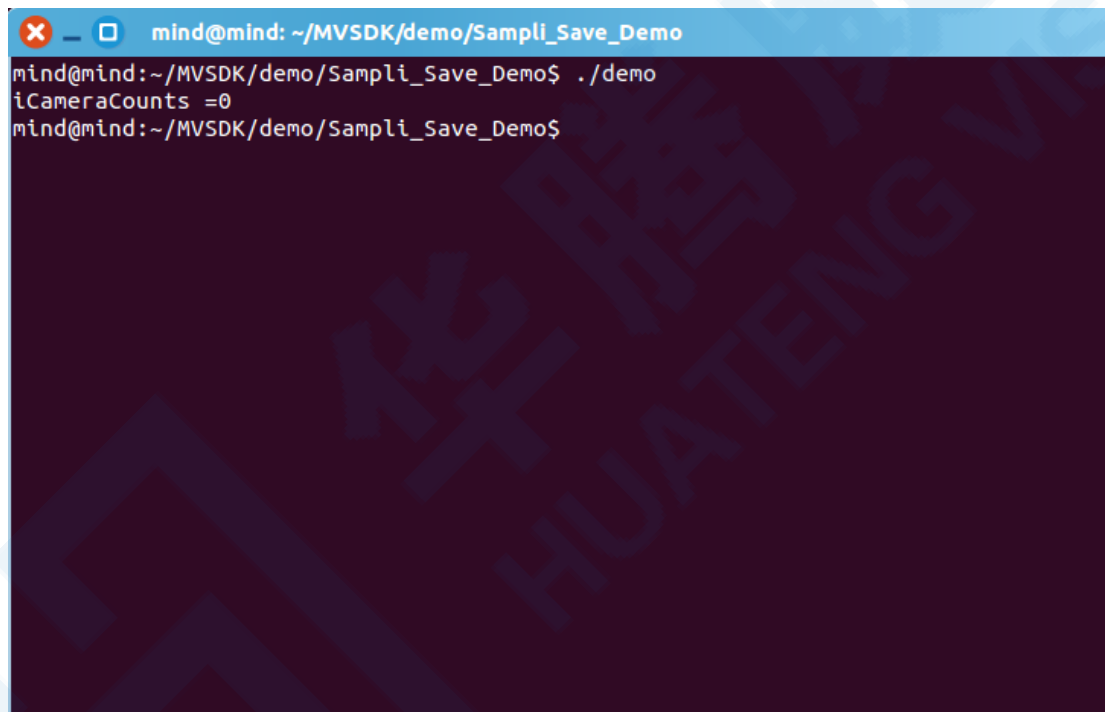
arm 32 hard floating point system: sudo cp MVSDK/lib/arm/libMVSDK.so /lib

arm 32 soft floating point system: sudo cp MVSDK/lib/arm_softfp/libMVSDK.so /lib

arm 64-bit system: sudo cp MVSDK/lib/arm64/libMVSDK.so /lib

2. The camera cannot be found when running the Demo

The number of cameras is 0

A terminal window titled 'mind@mind: ~/MVSDK/demo/Sampli_Save_Demo' shows the execution of a program. The user runs './demo' and the output is 'iCameraCounts =0'. The terminal background is dark purple with a large, faint watermark of the Chinese characters '华腾视觉' (HuaTeng Vision) and the English text 'HUAATENG VISION'.

1.Network port

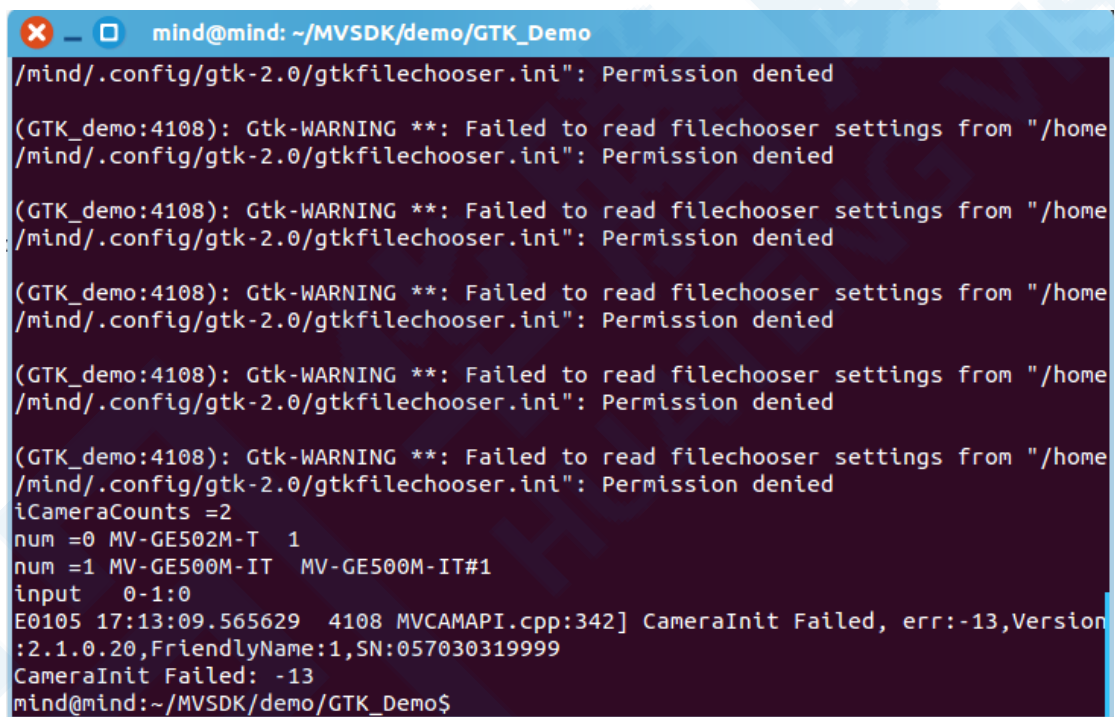
1. Turn off firewall sudo ufw disable
2. 2. Configure the network card IP of the linux computer as a fixed IP
3. Edit/etc/sysctl.d/10-network-security.conf Document, Set rp_filter to 0,
4. As follows: # Turn on Source Address Verification in all interfaces to
prevent some spoofing attacks.
net.ipv4.conf.default.rp_filter=0
net.ipv4.conf.all.rp_filter=0
Restart the computer to take effect

2. USB

1. **Lsusb** Check if the driver is loaded successfully f622:da01 The beginning of f622 means that the Huateng Vision camera has successfully loaded the driver
2. **sudo** Run
3. Try to change other USB ports
4. In a virtual machine, when using a USB3.0 camera, the USB controller sets the USB3.0 mode

3. Common errors when opening the camera

Return -13, the camera has been turned on, or the IP of the network card and the IP of the camera are not in the same network segment. Try to modify the camera IP, see for details [Camera IP Configuration](#).



```
mind@mind: ~/MVSDK/demo/GTK_Demo
/mind/.config/gtk-2.0/gtkfilechooser.ini": Permission denied
(GTK_demo:4108): Gtk-WARNING **: Failed to read filechooser settings from "/home/mind/.config/gtk-2.0/gtkfilechooser.ini": Permission denied
(GTK_demo:4108): Gtk-WARNING **: Failed to read filechooser settings from "/home/mind/.config/gtk-2.0/gtkfilechooser.ini": Permission denied
(GTK_demo:4108): Gtk-WARNING **: Failed to read filechooser settings from "/home/mind/.config/gtk-2.0/gtkfilechooser.ini": Permission denied
(GTK_demo:4108): Gtk-WARNING **: Failed to read filechooser settings from "/home/mind/.config/gtk-2.0/gtkfilechooser.ini": Permission denied
(GTK_demo:4108): Gtk-WARNING **: Failed to read filechooser settings from "/home/mind/.config/gtk-2.0/gtkfilechooser.ini": Permission denied
iCameraCounts =2
num =0 MV-GE502M-T 1
num =1 MV-GE500M-IT MV-GE500M-IT#1
input 0-1:0
E0105 17:13:09.565629 4108 MVCAMAPI.cpp:342] CameraInit Failed, err:-13,Version:2.1.0.20,FriendlyName:1,SN:057030319999
CameraInit Failed: -13
mind@mind:~/MVSDK/demo/GTK_Demo$
```

4. Camera IP Configuration

1. QT tool to modify IP (only supports x64)

Toolpath: [tools/QGeConfigTools/QGeConfigTools](#)

1. Automatically update IP

Select the camera to be modified, select DHCP mode, tick Save configuration, click the button to automatically update IP address, the IP modification is successful

网口相机IP配置工具

相机列表

eth0

- XG2500GM 172.31.1.247[DC-A5-01-88-C5-7B]
- GE500M 192.168.1.19[DC-A5-01-5D-EC-2F]**
- Camera MV-GE630M 172.31.1.221[DC-A5-01-63-C7-16]

eth1

本机信息

适配器: eth0

IP: 172 . 31 . 1 . 94

子网掩码: 255 . 255 . 255 . 0

网关: 172 . 31 . 1 . 1

网卡修改: [修改](#)

相机信息

名称: [设置](#)

类型: GE500M

版本: V3.2.136-3.4.108

序列号: AS057030310174

网卡地址: DC-A5-01-5D-EC-2F

IP: 192.168.1.19

子网掩码: 255.255.255.0

网关: 0.0.0.0

网口速度: 1000M

状态: IDLE

IP设置

IP: 192 . 168 . 1 . 19

子网掩码: 255 . 255 . 255 . 0

网关: 0 . 0 . 0 . 0

设置模式: ☐ 静态 ☒ DHCP

☒ 保存设置 [更新IP地址](#) [自动更新IP地址](#)

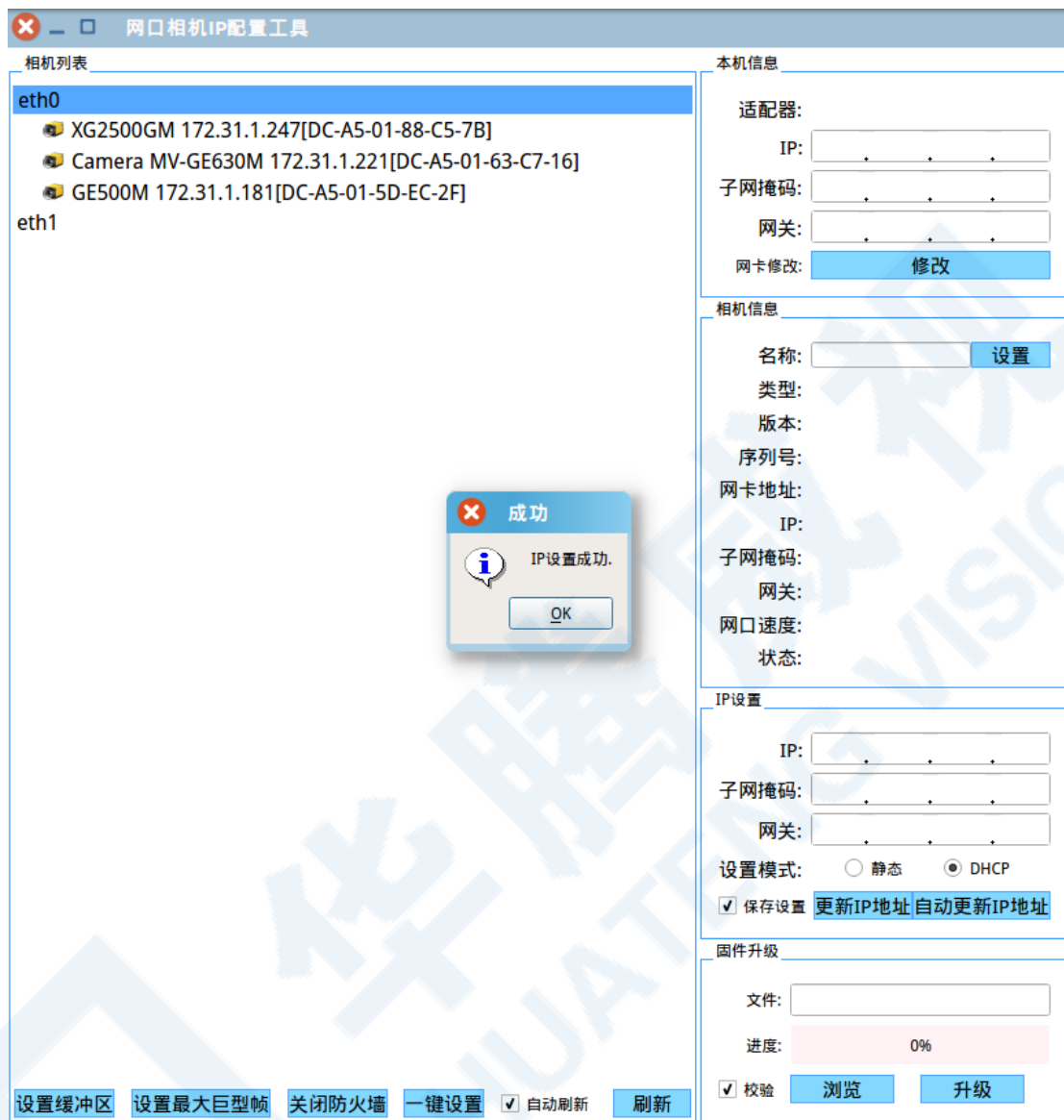
固件升级

文件:

进度: 0%

☒ 校验 [浏览](#) [升级](#)

设置缓冲区 设置最大巨型帧 关闭防火墙 一键设置 ☒ 自动刷新 [刷新](#)



2. Update IP manually

If you need to modify it to a specific IP, select the static mode, enter the IP address, subnet mask, gateway, tick Save the configuration, click the Update IP Address button, and the IP modification is successful.

网口相机IP配置工具

相机列表

eth0

- XG2500GM 172.31.1.247[DC-A5-01-88-C5-7B]
- Camera MV-GE630M 172.31.1.221[DC-A5-01-63-C7-16]
- GE500M 172.31.1.181[DC-A5-01-5D-EC-2F]**

eth1

本机信息

适配器: eth0

IP: 172 . 31 . 1 . 94

子网掩码: 255 . 255 . 255 . 0

网关: 172 . 31 . 1 . 1

网卡修改: [修改](#)

相机信息

名称: [设置](#)

类型: GE500M

版本: V3.2.136-3.4.108

序列号: AS057030310174

网卡地址: DC-A5-01-5D-EC-2F

IP: 172.31.1.181

子网掩码: 255.255.255.0

网关: 172.31.1.1

网口速度: 1000M

状态: IDLE

IP设置

IP: 172 . 31 . 1 . 182

子网掩码: 255 . 255 . 255 . 0

网关: 172 . 31 . 1 . 1

设置模式: ☒ 静态 ☐ DHCP

☒ 保存设置 [更新IP地址](#) [自动更新IP地址](#)

固件升级

文件:

进度: 0%

☒ 校验 [浏览](#) [升级](#)

[设置缓冲区](#) [设置最大巨型帧](#) [关闭防火墙](#) [一键设置](#) ☒ 自动刷新 [刷新](#)



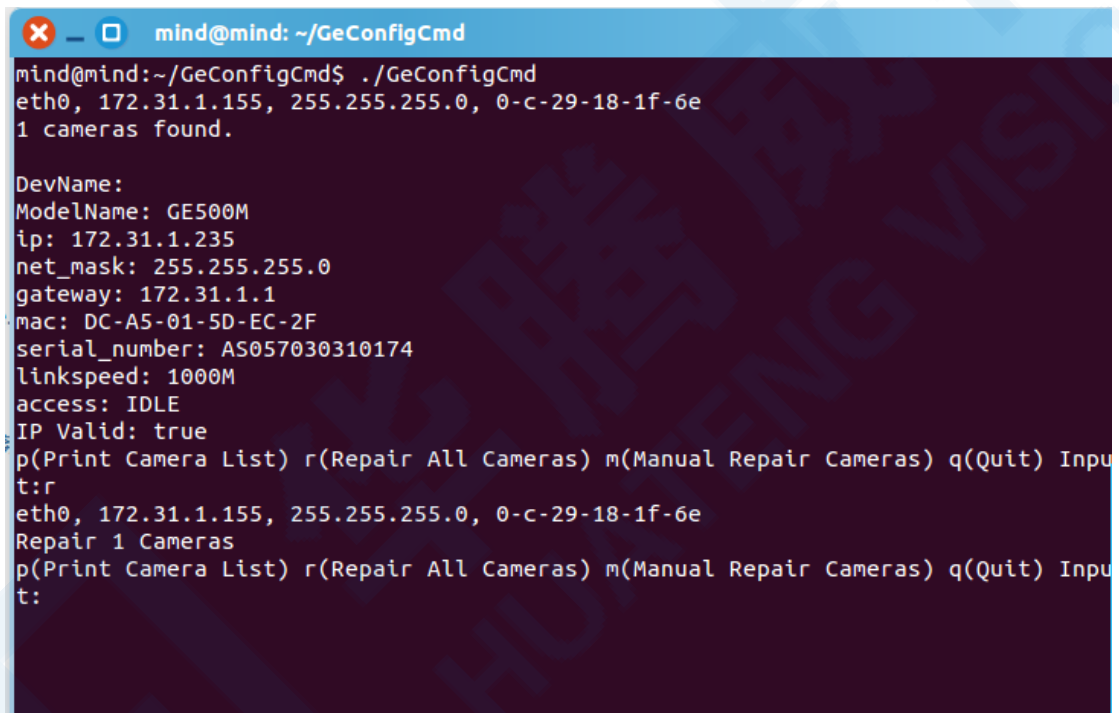
2. Command line tool to modify IP (support x86, x64, arm, arm64)

Toolpath:

x86: tools/GeConfigCmd/x86/GeConfigCmd
x64: tools/GeConfigCmd/x64/GeConfigCmd
arm: tools/GeConfigCmd/arm/GeConfigCmd
arm64: tools/GeConfigCmd/arm64/GeConfigCmd

Run the tool and enter r to automatically modify the IP

`./GeConfigCmd`



```
mind@mind: ~/GeConfigCmd
mind@mind:~/GeConfigCmd$ ./GeConfigCmd
eth0, 172.31.1.155, 255.255.255.0, 0-c-29-18-1f-6e
1 cameras found.

DevName:
ModelName: GE500M
ip: 172.31.1.235
net_mask: 255.255.255.0
gateway: 172.31.1.1
mac: DC-A5-01-5D-EC-2F
serial_number: AS057030310174
linkspeed: 1000M
access: IDLE
IP Valid: true
p(Print Camera List) r(Repair All Cameras) m(Manual Repair Cameras) q(Quit) Input:
t:r
eth0, 172.31.1.155, 255.255.255.0, 0-c-29-18-1f-6e
Repair 1 Cameras
p(Print Camera List) r(Repair All Cameras) m(Manual Repair Cameras) q(Quit) Input:
t:
```

3. Compile GTK_Demo

1. Install the GTK library

Install the GTK library, clean the project, compile with make, run ./Gtk_Demo, enter 0 to open the first camera

`sudo apt-get install libgtk2.0-dev`

`cd MVSDK/demo/GTK_Demo`

`#enter`

the

directory

`make clean`

`#clean`

project

`make`

`#compile GTKdemo`

```
mind@mind: ~/MVSDK/demo/GTK_Demo
mind@mind:~/MVSDK/demo/GTK_Demo$ sudo apt-get install libgtk-3-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  account-plugin-windows-live gtk3-engines-unico libntdb1 libqpdf13
  libupstart1 python-ntdb
Use 'apt-get autoremove' to remove them.
Suggested packages:
  libgtk-3-doc
The following NEW packages will be installed:
  libgtk-3-dev
0 upgraded, 1 newly installed, 0 to remove and 212 not upgraded.
Need to get 0 B/807 kB of archives.
After this operation, 9859 kB of additional disk space will be used.
WARNING: The following packages cannot be authenticated!
  libgtk-3-dev
Install these packages without verification? [y/N] y
Selecting previously unselected package libgtk-3-dev.
(Reading database ... 216235 files and directories currently installed.)
Preparing to unpack .../libgtk-3-dev_3.10.8-0ubuntu1.6_amd64.deb ...
Unpacking libgtk-3-dev (3.10.8-0ubuntu1.6) ...
Setting up libgtk-3-dev (3.10.8-0ubuntu1.6) ...
mind@mind:~/MVSDK/demo/GTK_Demo$
```

```

mind@mind: ~/MVSDK/demo/GTK_Demo
size [-Wint-to-pointer-cast]
  ECT(radio_speed[i]),"released",G_CALLBACK(on_radio_speed_clicked),(gpointer)i);

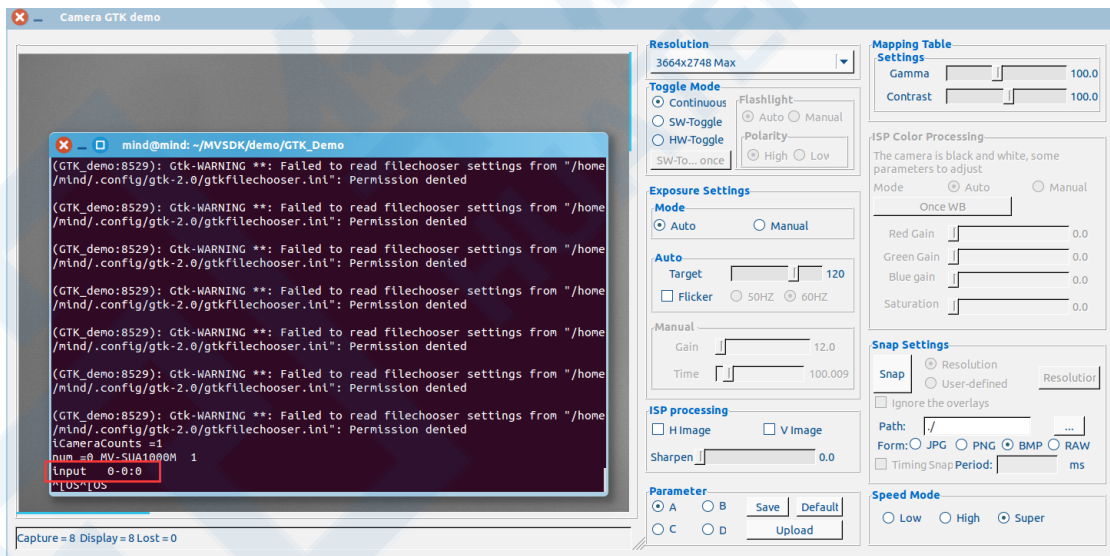
/usr/include/glib-2.0/gobject/gsignal.h:472:73: note: in definition of macro 'g_
signal_connect'
  g_signal_connect_data ((instance), (detailed_signal), (c_handler), (data), NULL
mv -f .deps/interface.Tpo .deps/interface.Po
g++ -DHAVE_CONFIG_H -I. -I./inc -I../include -pthread -I/usr/include/gtk-2.
0 -I/usr/lib/x86_64-linux-gnu/gtk-2.0/include -I/usr/include/gio-unix-2.0/ -I/us
r/include/cairo -I/usr/include/pango-1.0 -I/usr/include/atk-1.0 -I/usr/include/c
airo -I/usr/include/pixman-1 -I/usr/include/libpng12 -I/usr/include/gdk-pixbuf-2
.0 -I/usr/include/libpng12 -I/usr/include/pango-1.0 -I/usr/include/harfbuzz -I/u
sr/include/pango-1.0 -I/usr/include/glib-2.0 -I/usr/lib/x86_64-linux-gnu/glib-2.
0/include -I/usr/include/freetype2 -g -O2 -MT support.o -MD -MP -MF .deps/su
pport.Tpo -c -o support.o `test -f './src/support.cpp' || echo './'`./src/suppor
t.cpp
mv -f .deps/support.Tpo .deps/support.Po
g++ -g -O2 -L/lib -o GTK_demo GTK_demo.o callbacks.o Demo.o interface.o suppor
t.o -lgtk-x11-2.0 -lgdk-x11-2.0 -lpangocairo-1.0 -latk-1.0 -lcairo -lgdk_pixbuf-
2.0 -lgio-2.0 -lpangoft2-1.0 -lpango-1.0 -lgobject-2.0 -lglib-2.0 -lfonconfig -
lfreetype -lpthread -lrt -lgthread-2.0 -lmvSDK -lthread
make[1]: Leaving directory `/home/mind/MVSDK/demo/GTK_Demo'
mind@mind:~/MVSDK/demo/GTK_Demo$

```

No obvious errors, you can directly execute GTK_DEMO

sudo ./GTK_demo

Select the first camera, press 0 to start by default, enter 0 and press
Enter to open the first camera



4. Compile multicamera_demo

Clean the project, make compile, run the program, open 2 cameras

cd MVSDK/demo/multicamera_demo

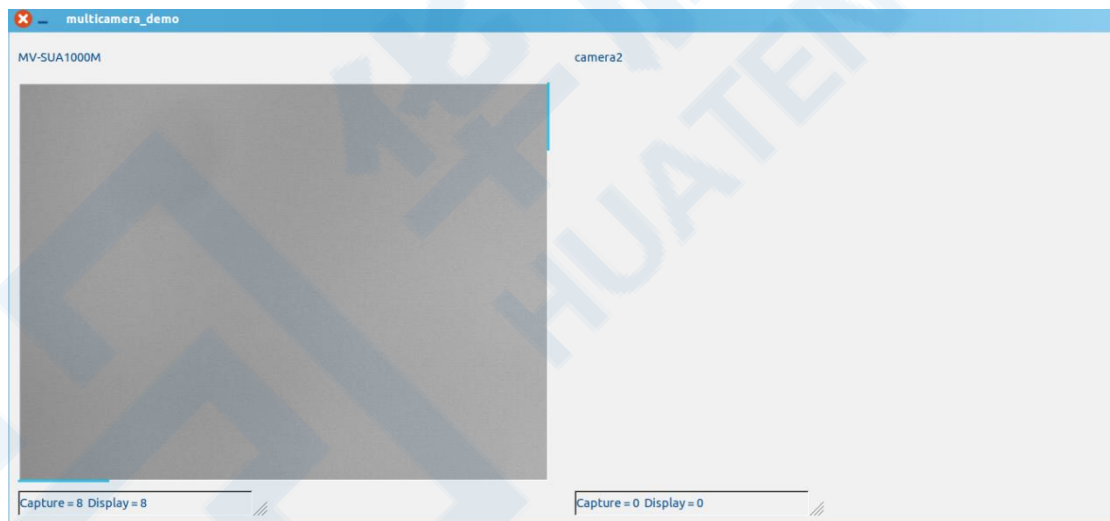
make clean

make

```
mind@mind: ~/MVSDK/demo/multicamera_demo
from /usr/include/gtk-2.0/gdk/gdk.h:32,
from /usr/include/gtk-2.0/gtk/gtk.h:32,
from ./src/interface.cpp:2:
/usr/include/glib-2.0/glib/gmacros.h:229:0: note: this is the location of the previous definition
#define FALSE (0)
^
mv -f .deps/interface.Tpo .deps/interface.Po
g++ -DHAVE_CONFIG_H -I. -I./inc -I../include -pthread -I/usr/include/gtk-2.0 -I/usr/lib/x86_64-linux-gnu/gtk-2.0/include -I/usr/include/gio-unix-2.0/ -I/usr/include/cairo -I/usr/include/pango-1.0 -I/usr/include/atk-1.0 -I/usr/include/cairo -I/usr/include/pixman-1 -I/usr/include/libpng12 -I/usr/include/gdk-pixbuf-2.0 -I/usr/include/libpng12 -I/usr/include/pango-1.0 -I/usr/include/harfbuzz -I/usr/include/pango-1.0 -I/usr/include/glib-2.0 -I/usr/lib/x86_64-linux-gnu/glib-2.0/include -I/usr/include/freetype2 -g -O2 -MT support.o -MD -MP -MF .deps/support.Tpo -c -o support.o `test -f './src/support.cpp' || echo './src/support.cpp'
mv -f .deps/support.Tpo .deps/support.Po
g++ -g -O2 -L/lib -o multicamera_demo multicamera_demo.o callbacks.o Demo.o interface.o support.o -lgtk-x11-2.0 -lgdk-x11-2.0 -lpangocairo-1.0 -latk-1.0 -lcairo -lgdk_pixbuf-2.0 -lgio-2.0 -lpangoft2-1.0 -lpango-1.0 -lgobject-2.0 -lglib-2.0 -lfontconfig -lfreetype -lpthread -lrt -lgthread-2.0 -lMVSDK -lpthread
make[1]: Leaving directory `/home/mind/MVSDK/demo/multicamera_demo'
mind@mind:~/MVSDK/demo/multicamera_demo$
```

No obvious errors, run the program

`sudo ./multicamera_demo`



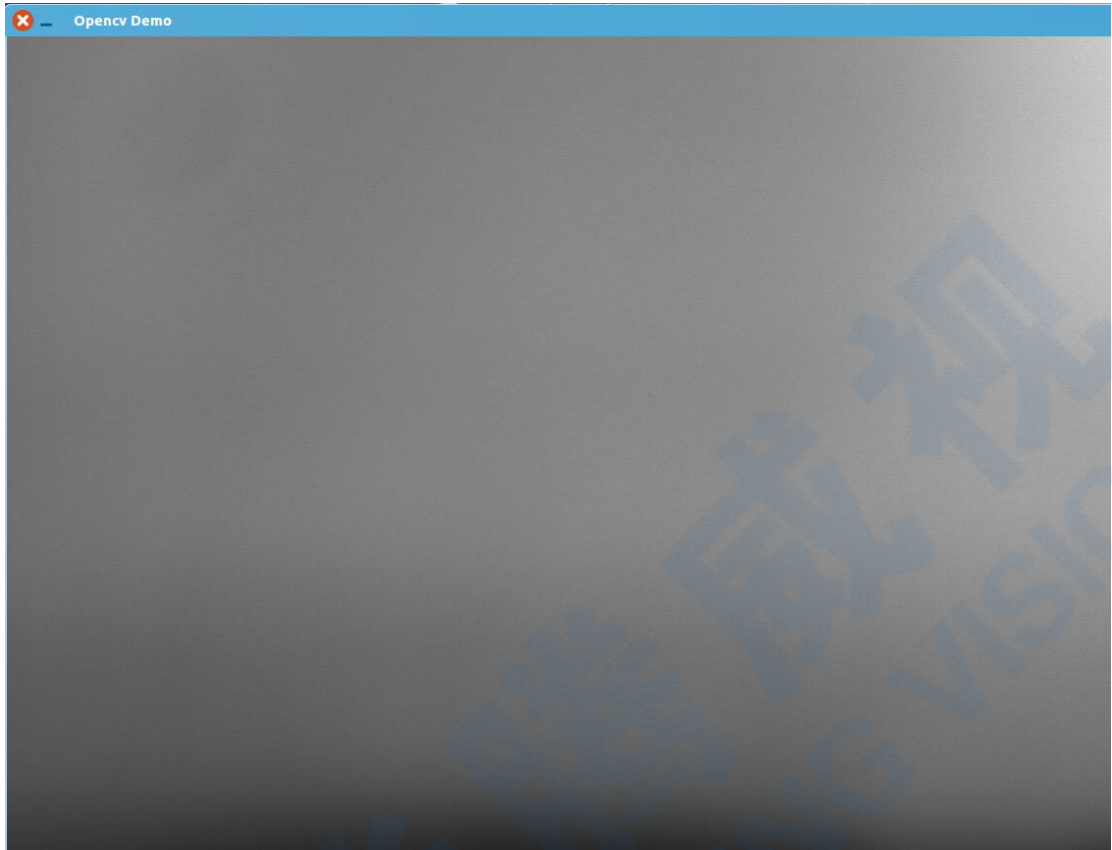
5. Compile OpenCv

Install opencv first, or download the source code from the official website, compile it, enter the directory, compile, and run

`cd MVSDK/demo/OpenCv`

`make`

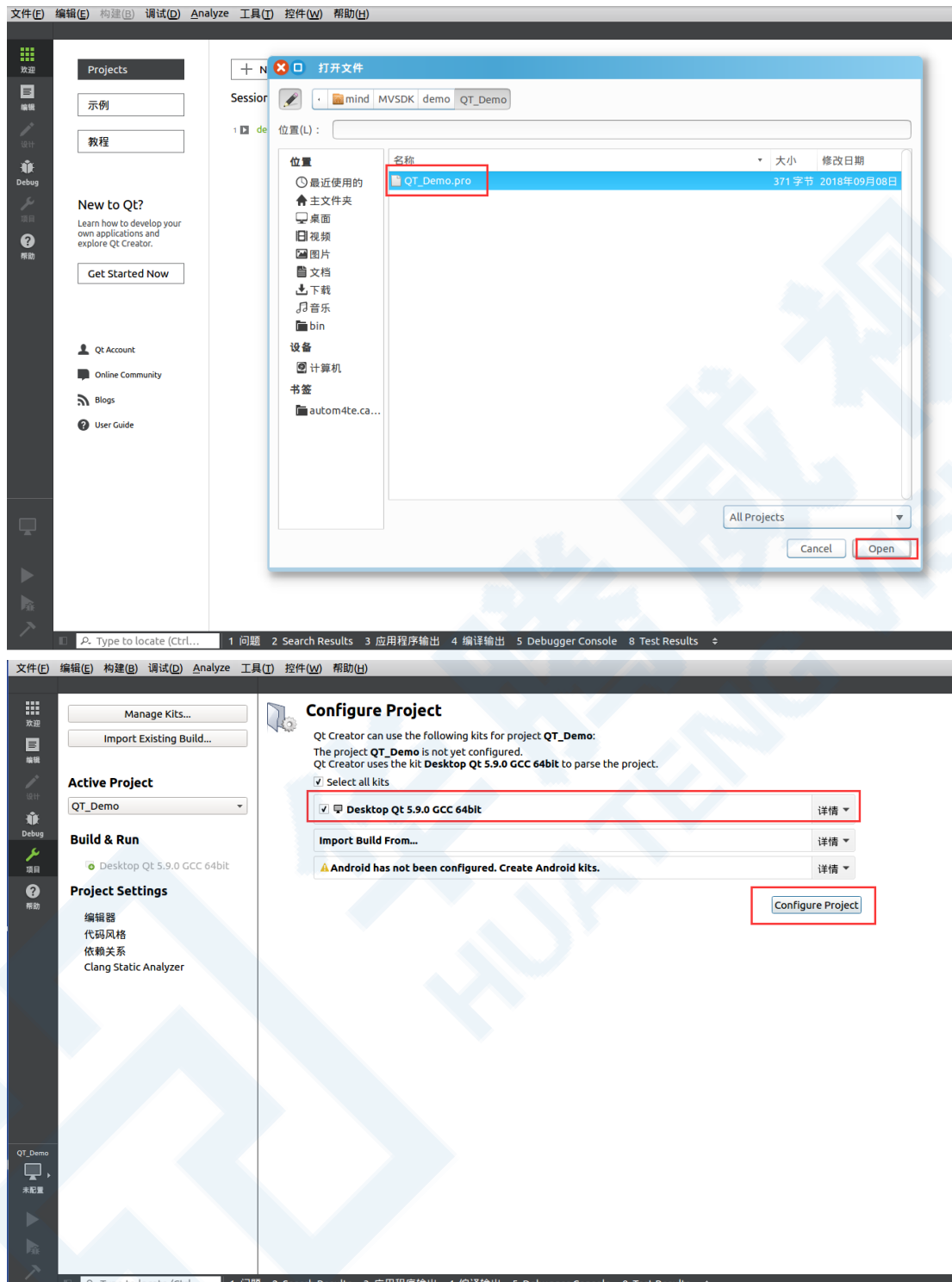
`sudo ./main` `#Run`



6. compile QT_Demo4

1. You need to install Qt first

Open Qt, open the project, select the pro file, open, select the corresponding builder
(QT_Demo4 changed to QT5)

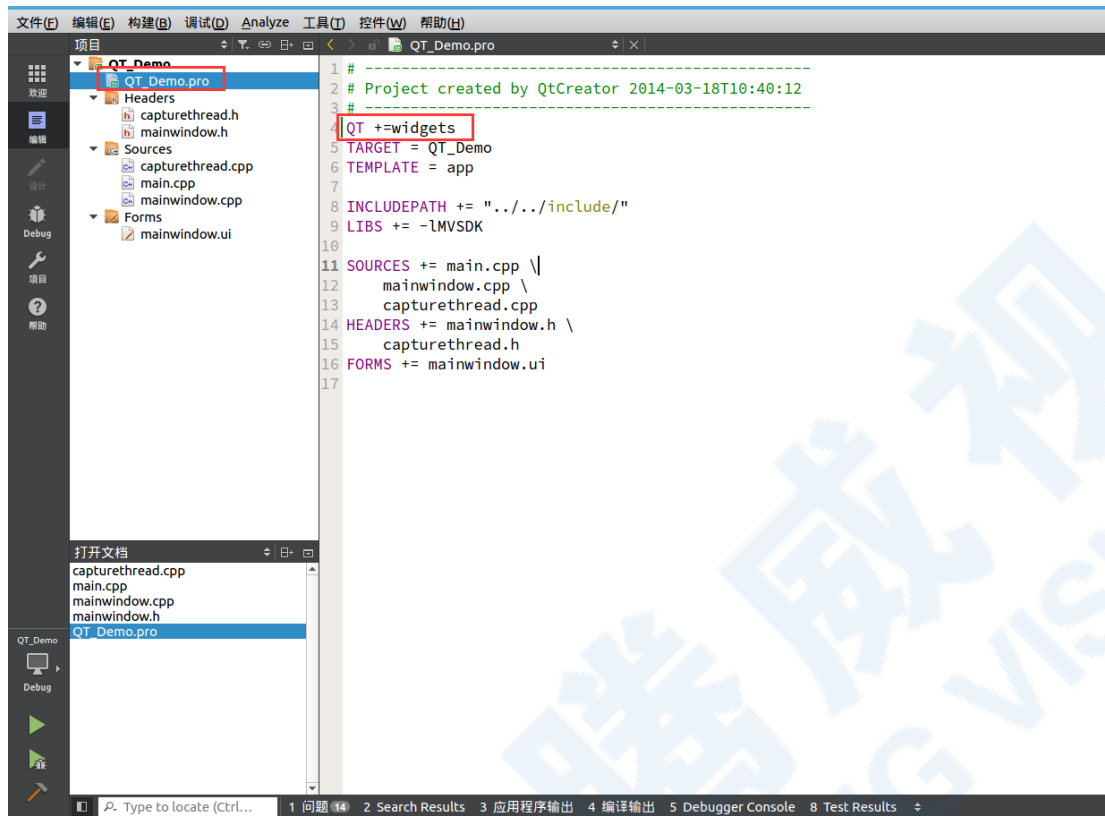


Start building

Pro file added QT += widgets

INCLUDEPATH += ../../include #Revise

LIBS += -L/lib -lMVSDK #Revise



if it appears `#include <QtGui/QApplication>` Error change to `#include <QApplication>`

There are a lot of errors, add a header file reference

mainwindow.h Add reference to header file

`#include <QLabel>`

`#include <QGraphicsScene>`

`#include <QGraphicsPixmapItem>`

`#include <QRadioButton>`

`#include <QDesktopWidget>`

`#include <QFileDialog>`

`#include <QTextCodec>`

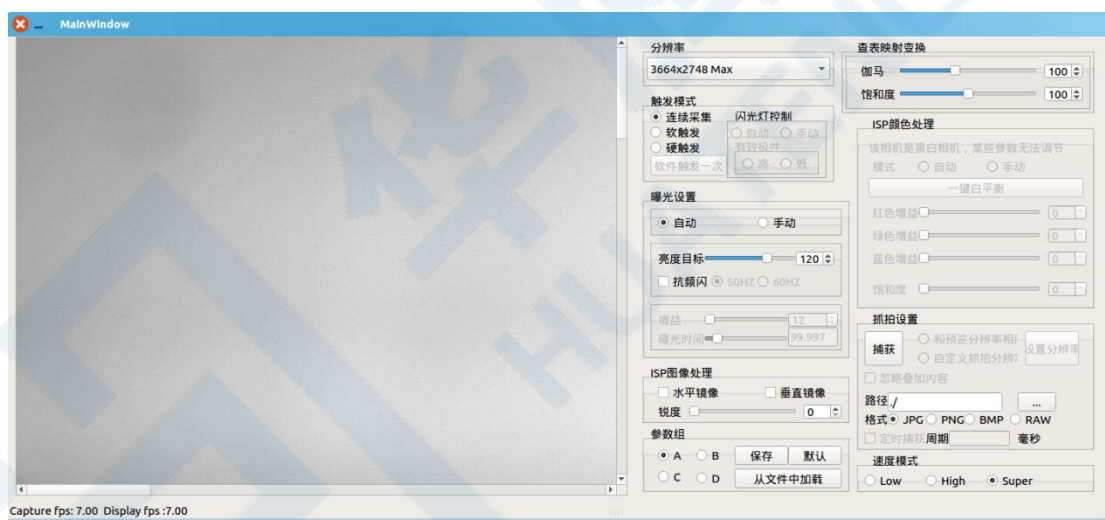
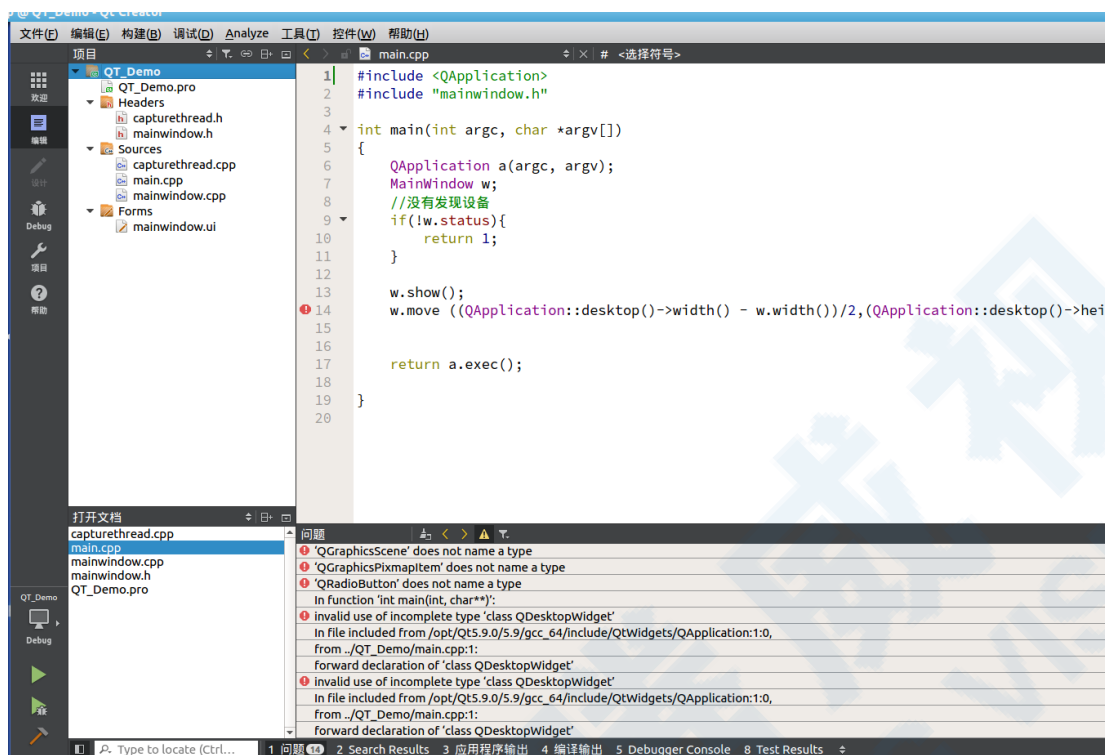
`#include <QMessageBox>`

`QTextCodec::setCodecForCStrings(QTextCodec::codecForName("GB2312"));` Delete

`fileDialog->setFilterst(tr("config Files (*.config)"));`

Change to

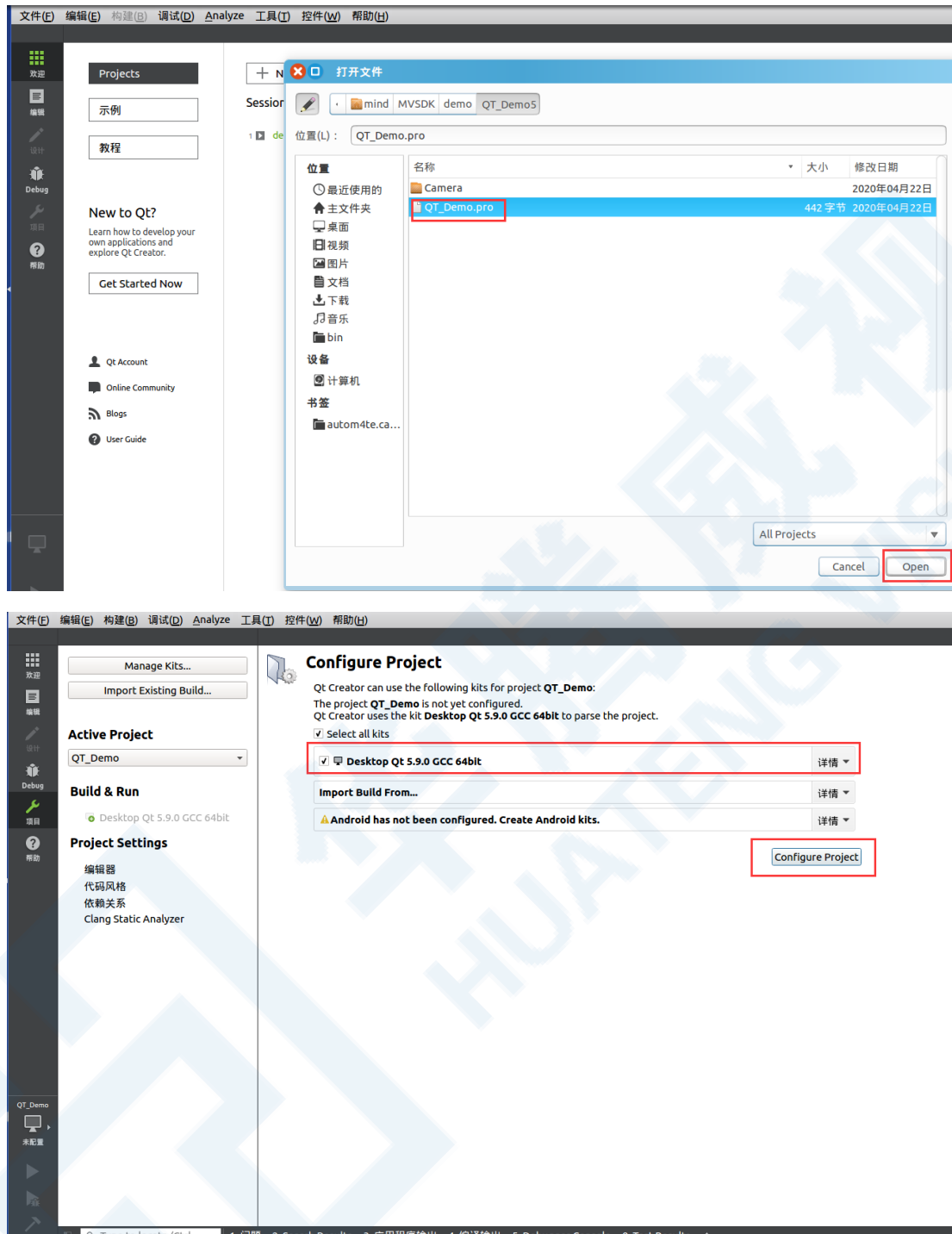
`fileDialog->setNameFilters(QStringList()<<tr("config Files (*.config)"));`

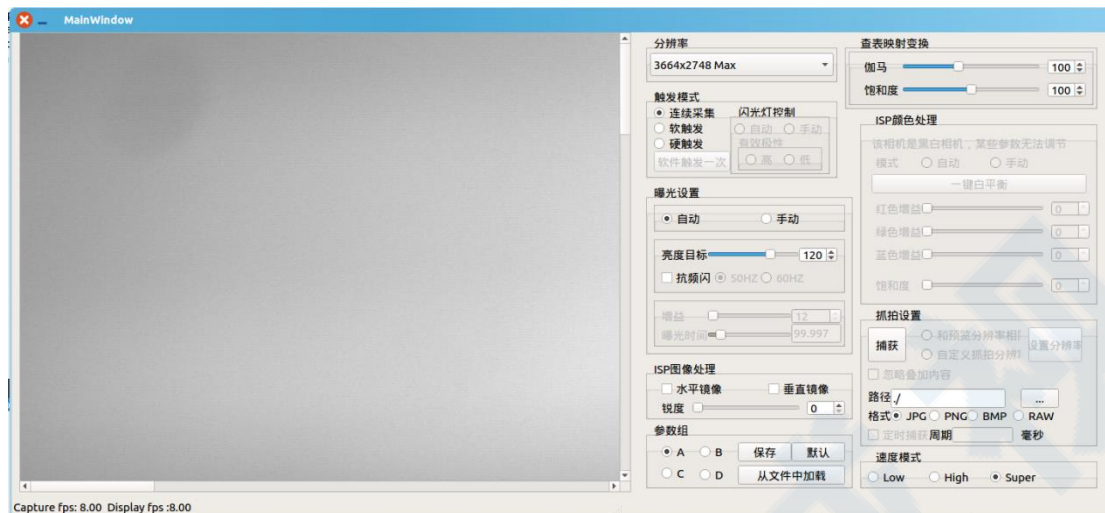


7. Compile QT_Demo5

1. You need to install Qt first

Open Qt, open the project, select the pro file, open, select the corresponding builder, compile and run, you can





8. Compile Sampli_Save_Demo

Enter the directory, clean the project, make compile `./demo` run, enter 0, open the 0th camera, save an image

```
cd MVSDK/demo/Sampli_Save_Demo
```

```
make clean
```

```
make
```

```
sudo ./demo
```

```
mind@mind: ~/MVSDK/demo/Sampli_Save_Demo
make all-am
make[1]: Entering directory `/home/mind/MVSDK/demo/Sampli_Save_Demo'
g++ -DHAVE_CONFIG_H -I. -I../inc -I../include -g -O2 -MT main.o -MD -MP -MF .deps/main.Tpo -c -o main.o main.cpp
main.cpp: In function 'int main()':
main.cpp:51:22: warning: ignoring return value of 'int scanf(const char*, ...)',
declared with attribute warn_unused_result [-Wunused-result]
    scanf("%d", &num);
                        ^
mv -f .deps/main.Tpo .deps/main.Po
g++ -g -O2 -L/lib -o demo main.o -lmvSDK -lrt -lpthread -lpthread
make[1]: Leaving directory `/home/mind/MVSDK/demo/Sampli_Save_Demo'
mind@mind:~/MVSDK/demo/Sampli_Save_Demo$ ./demo
tcameracounts =2
num =0 MV-SUA1000M 1
num =1 MV-GE500M-IT MV-GE500M-IT#0
input 0-1:0
you input num 0
CameraInit iStatus =0
CameraGetCapability
CameraPlay
CameraSetIspOutFormat
end
mind@mind:~/MVSDK/demo/Sampli_Save_Demo$
```

