

1 前言

镜像已经做好了，传到百度网盘中了（请大家及时保存，不定期删除！）

链接: <https://pan.baidu.com/s/1fhiX86L8iL8tsLbsiVa6Wg> 密码: e64s

镜像使用说明：

- **root** 密码 **111**
- 依赖库编译在 **Downloads** 文件夹下
- 编译好的 **NCNN** 项目在 **work** 目录下

SD 卡要求：至少 **16GB**，板卡型号为树莓派 **3B+**（其他型号未知）

2 板卡介绍

本系列教程采用树莓派 3B+ 开发板：

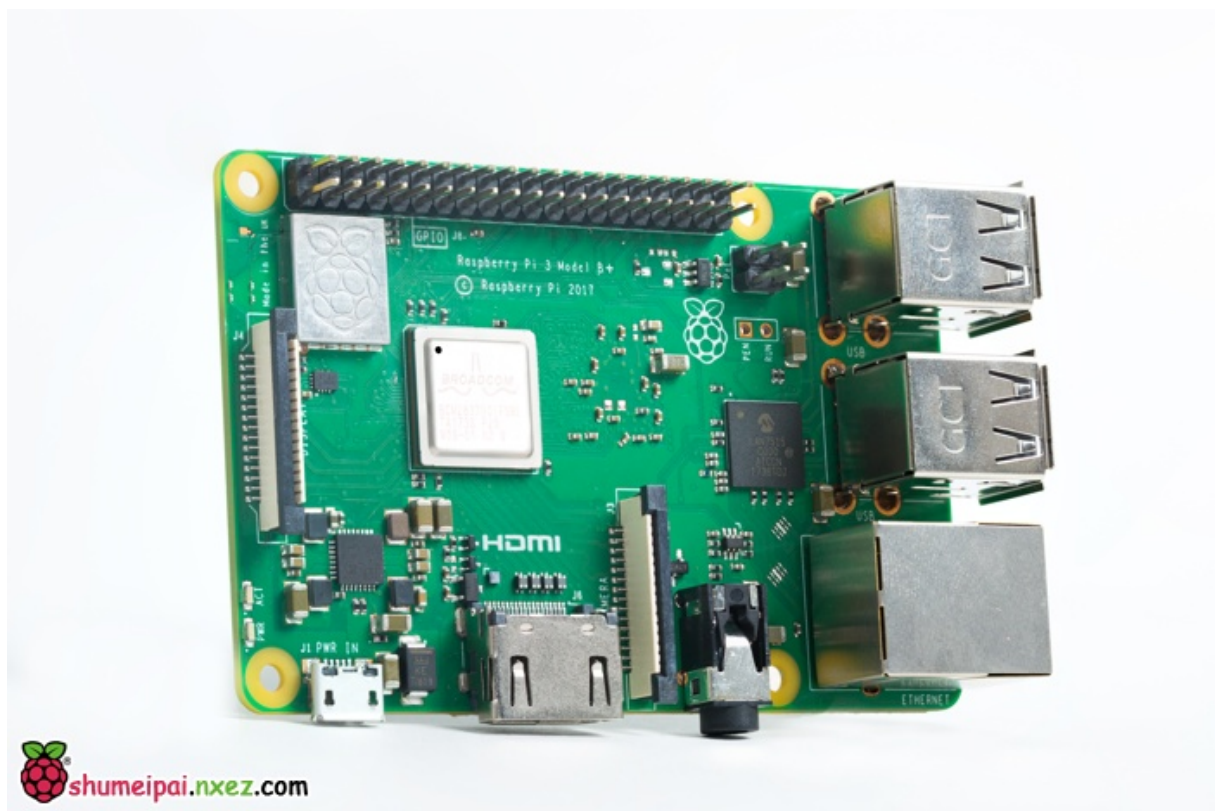


Figure 1: 开发板示意图

板卡特点：

- 1.4GHz 64 位 4 核 ARM Cortex-A53 CPU
- 双频 802.11ac 无线网卡和蓝牙 4.2
- 更快的以太网（千兆以太网 over USB 2.0）
- 1G LPDDR2
- PoE 支持（Power-over-Ethernet，with PoE HAT）
- 改进 PXE 网络与 USB 大容量存储启动

3 系统安装

3.1 工具

- SDFormatter(格式化 SD 卡)
- win32diskimager（为 SD 卡烧写程序）

3.2 安装步骤

首先用 SDFormatter 将插入的 SD 卡格式化。然后用 win32diskimager 找到对应的镜像烧写到 SD 卡中（原始镜像采用 2018-04-18-raspbian-stretch 树莓派官方系统）

4 更换源

4.1 一步操作

直接执行以下两步，即可替换将官方默认软件源替换为中科大镜像源

```
1 $ sudo sed -i 's#://mirrordirector.raspbian.org#s://mirrors.ustc.edu.cn
   /raspbian#g' /etc/apt/sources.list
2 $ sudo sed -i 's#://archive.raspberrypi.org/debian#s://mirrors.ustc.edu
   .cn/archive.raspberrypi.org#g' /etc/apt/sources.list.d/raspi.list
```

或换为清华镜像源

```
1 $ sudo sed -i 's#://mirrordirector.raspbian.org#s://mirrors.tuna.
   tsinghua.edu.cn/raspbian#g' /etc/apt/sources.list
2 $ sudo sed -i 's#://archive.raspberrypi.org/debian#s://mirrors.tuna.
   tsinghua.edu.cn/raspberrypi#g' /etc/apt/sources.list.d/raspi.list
```

4.2 或手动修改源

- 第一步：修改sources.list

```
1 $ sudo vim /etc/apt/sources.list
```

进入之后，屏蔽掉其他的源，输入以下源：

```
1 $ deb http://mirrors.tuna.tsinghua.edu.cn/raspbian/raspbian/  
    stretch main contrib non-free rpi  
2 $ deb-src http://mirrors.tuna.tsinghua.edu.cn/raspbian/raspbian/  
    stretch main contrib non-free rpi
```

- 第二步：修改raspi.list

```
1 $ sudo vim /etc/apt/sources.list.d/raspi.list
```

进入之后，屏蔽掉其他的源，输入以下源：

```
1 $ deb http://mirror.tuna.tsinghua.edu.cn/raspberrypi/ stretch main  
    ui  
2 $ deb-src http://mirror.tuna.tsinghua.edu.cn/raspberrypi/ stretch  
    main ui
```

- 第三步：修改完源后更新升级系统

```
1 $ sudo apt-get update  
2 $ sudo apt-get upgrade
```

5 安装 cmake 工具

5.1 安装 cmake

```
1 $ sudo apt-get install cmake
```

5.2 安装 cmake-gui

cmake-gui 是可视化的 cmake 工具，便于配置。

```
1 $ sudo apt-get install cmake-gui
```

6 安装 protobuf

下载 protobuf-2.6.1.tar.gz 安装

```
1 $ #解压文件
2 $ tar zxvf protobuf-2.6.1.tar.gz
3 $ #进入文件
4 $ cd protobuf-2.6.1/
5 $ # 配置
6 $ ./configure
7 $ # 编译（编译过程尽量只用一个核 不要加 j4,j3,j2）
8 $ make
9 $ # 编译检查
10 $ make check
11 $ # 安装
12 $ sudo make install
13 $ # 添加库路径 在/etc/ld.so.conf.d/目录下创建文件bprotobuf.conf文件
14 $ cd /etc/ld.so.conf.d/
15 $ sudo touch bprotobuf.conf
16 $ # vim打开bprotobuf.conf文件输入/usr/local/lib
17 $ sudo ldconfig
18 $ # 查看版本
19 $ protoc --version
```

7 安装 opencv3.4

7.1 预装依赖库

```
1 $ sudo apt-get install build-essential git cmake pkg-config -y
2 $ sudo apt-get install libjpeg8-dev -y
3 $ sudo apt-get install libtiff5-dev -y
4 $ sudo apt-get install libjasper-dev -y
5 $ sudo apt-get install libpng12-dev -y
6 $ sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev
   libv4l-dev -y
7 $ sudo apt-get install libgtk2.0-dev -y
8 $ sudo apt-get install libatlas-base-dev gfortran -y
9 $ sudo apt-get install qt5-default -y
```

安装 numpy（命令行），第一次使用 pip 安装时可能会比较慢，耐心等待

```
1 $ sudo pip3 install numpy
```

查看 numpy 的安装目录并记录（后面编译时需要 **PYTHON3_NUMPY_INCLUDE_DIRS** 路径，如果不带 **numpy** 编译可能会卡住）

```
1 $ python3
2 >> import numpy
3 >> numpy.__path__
4 >> quit()
```

7.2 下载并解压 OpenCV

然后下载 OpenCV 库和 Contrib 库（强烈建议在其他环境下载然后拷贝过来），如果需要其他版本，就修改后面的版本号

```
1 https://github.com/Itseez/opencv/archive/3.4.0.zip
2 https://github.com/Itseez/opencv_contrib/archive/3.4.0.zip
```

7.3 cmake 配置

解压后在 opencv-3.4.0 文件夹里创建 build 文件夹，然后在命令行里面 cd 到此文件夹，开始 cmake，以下内容为一行

```
1 $ cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local
   -D INSTALL_C_EXAMPLES=ON -D INSTALL_PYTHON_EXAMPLES=ON -D
   BUILD_EXAMPLES=ON -D WITH_LIBV4L=ON PYTHON3_EXECUTABLE=/usr/bin/
   python3.5 PYTHON_INCLUDE_DIR=/usr/include/python3.5 PYTHON_LIBRARY=/
   usr/lib/arm-linux-gnueabi/libpython3.5m.so
   PYTHON3_NUMPY_INCLUDE_DIRS=/home/pi/.local/lib/python3.5/site-
   packages/numpy/core/include ..
2
3 #备选编译方案
4 $ sudo aptitude search libgtk2.0-dev
5
6 $ cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local
   -D WITH_TBB=ON -D BUILD_NEW_PYTHON_SUPPORT=ON -D WITH_V4L=ON -D
   INSTALL_C_EXAMPLES=ON -D INSTALL_PYTHON_EXAMPLES=ON -D
   BUILD_EXAMPLES=ON -D WITH_QT=ON -D WITH_GTK=ON -D WITH_OPENGL=ON ..
```

7.4 编译安装

执行如下操作：

```
1 # 编译（编译过程尽量只用一个核 不要加 j4,j3,j2）
2 $ make
3 $ sudo make install
```

8 编译 NCNN

8.1 下载 NCNN

```
1 $ git clone https://github.com/Tencent/ncnn.github
```

8.2 需要安装的依赖库

- protobuf
- opencv

8.3 编译

```
1 $ cd ncnn
2 $ mkdir build
3 $ cd build
4 $ cmake ..
5 $ make
```

9 NCNN 测试

修改项目根目录下的CMakeLists.txt文件，定位到最后几行

```
#####  
  
add_subdirectory(examples)  
add_subdirectory(src)  
if(NOT ANDROID AND NOT IOS)  
#add_subdirectory(tools)  
endif()
```

Figure 2: CMakeLists.txt

```
1 $ cd <ncnn-root-dir>  
2 $ cd build  
3 $ cmake ..  
4 $ make  
5 $ cp examples/squeezenet_v1.1.param build/examples  
6 $ cp examples/squeezenet_v1.1.bin build/examples  
7 $ cd build/examples  
8 $ ./squeezenet cat.jpg
```

结果：

```
281 = 0.227178  
282 = 0.221584  
278 = 0.162667
```

Figure 3: 结果

10 调整 SD 卡分区

10.1 使用工具：

Linux Ubuntu gparted 分区工具

10.2 安装 gparted

```
1 $ sudo apt-get install gparted
```

10.3 调整分区容量大小

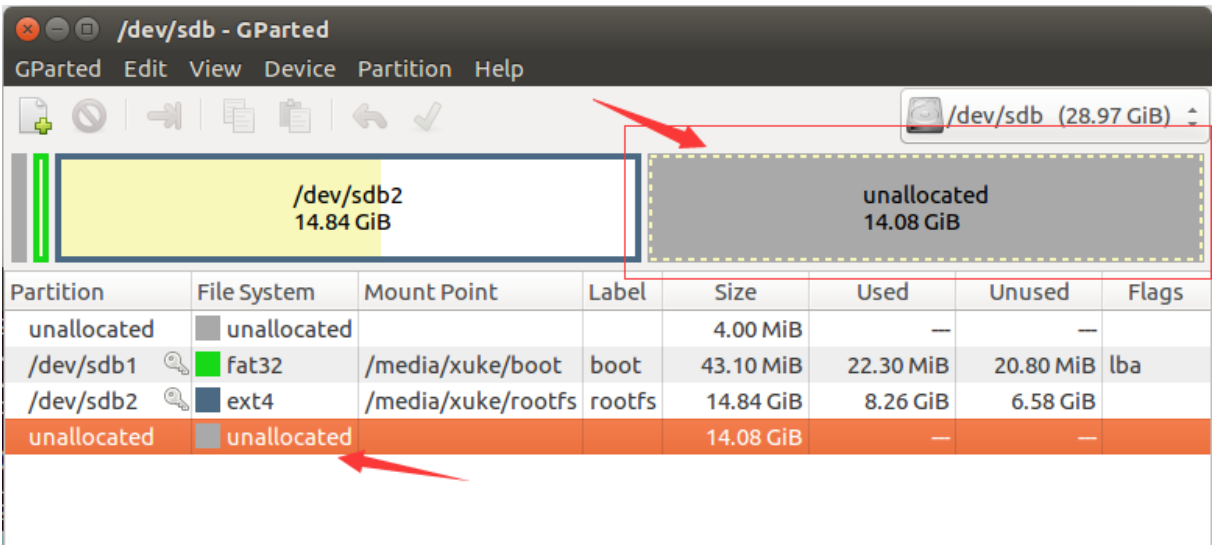


Figure 4: 步骤 1

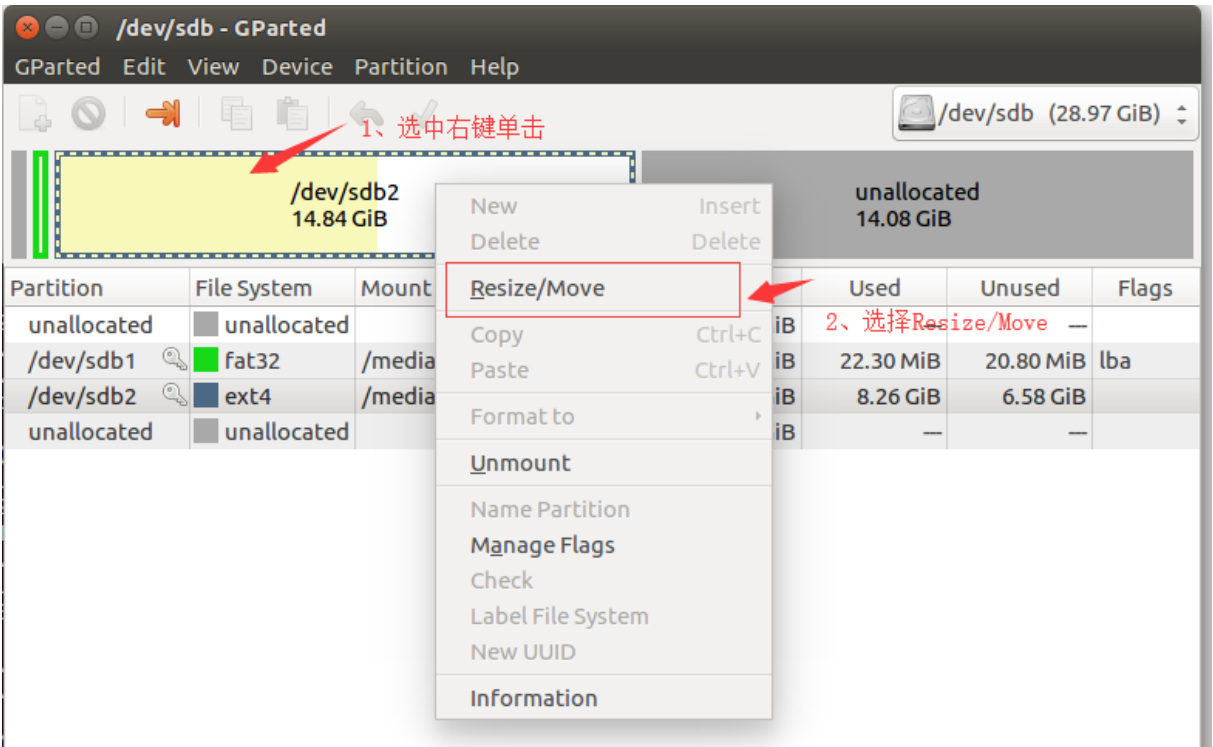


Figure 5: 步骤 2

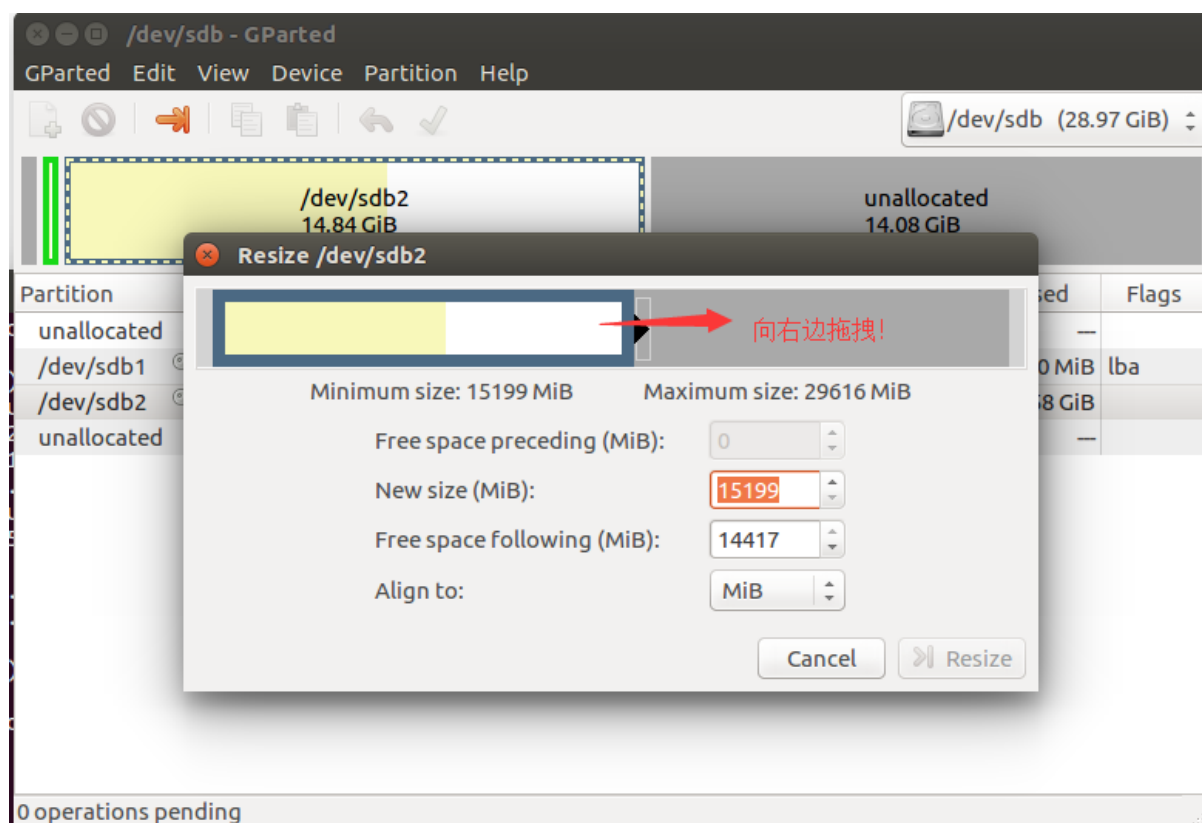


Figure 6: 步骤 3

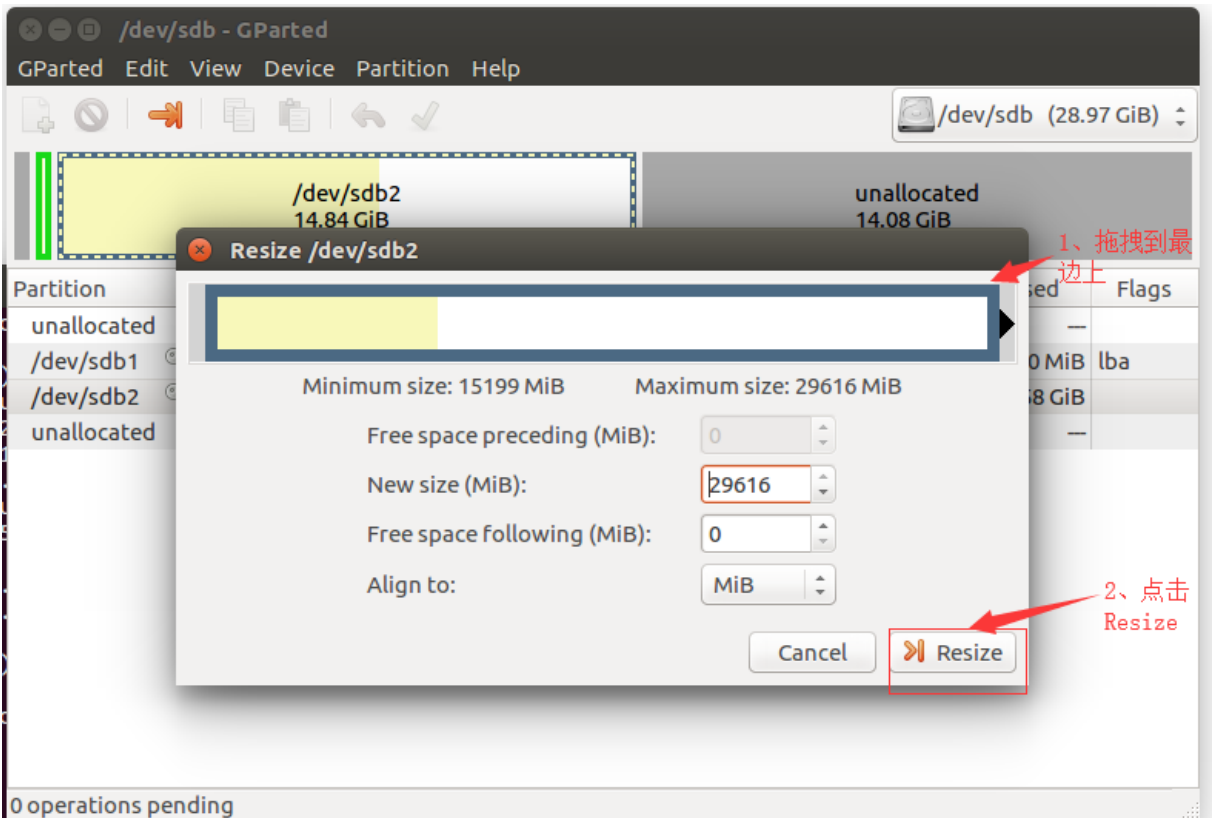


Figure 7: 步骤 4

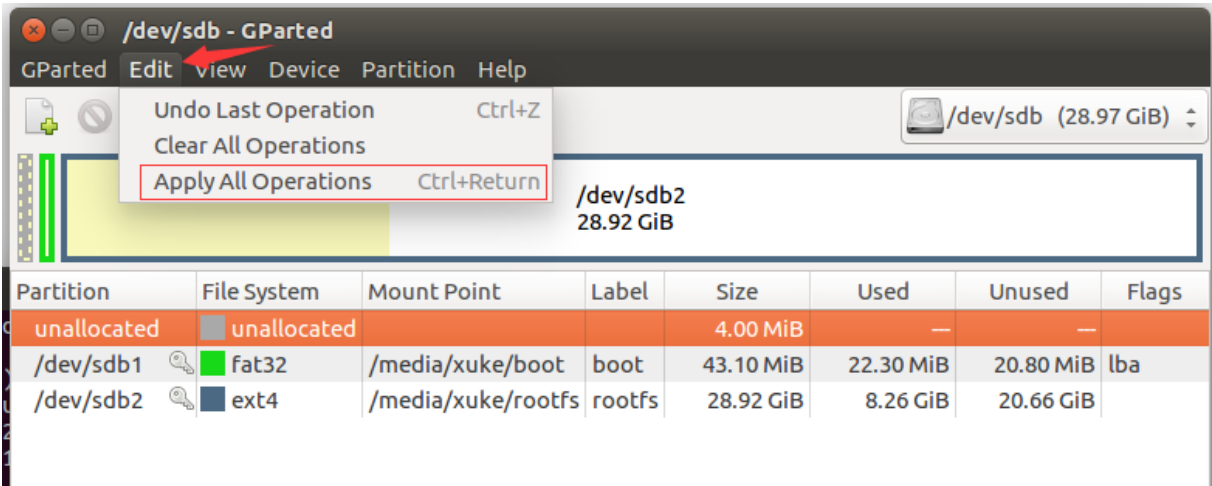


Figure 8: 步骤 5

11 给你个任务 o(￣▽￣)o

相关人工智能与异构计算的知识分享，欢迎关注我的公众号 AI 异构



Figure 9: 动动手指关注下吧！

12 参考

<https://www.jianshu.com/p/67b9e6ebf8a0>

<http://www.rabbit8.cn/609.html>

<https://linux.cn/article-8477-1.html>

<https://www.cnblogs.com/Pyrokin/p/8921285.html>

<https://stackoverflow.com/questions/28776053/opencv-gtk2-x-error>