RPI High Speed Camera micro-SD Images

Source Media

Sandisk microSDHC UHS-I 32 GB

Maximum speed: 100 MB/s, 667X

App performance: A1
Video speed class: V30

inxi

ID-1: /dev/mmcblk0 model: SM32G size: 29.72 GiB

fdisk -l

```
Disk /dev/mmcblk0: 29.7 GiB, 31914983424 bytes, 62333952 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x4b16a8ce

Device         Boot  Start      End  Sectors  Size Id Type
/dev/mmcblk0p1        8192   532479   524288  256M  c W95 FAT32 (LBA)
/dev/mmcblk0p2      532480 62333951 61801472 29.5G 83 Linux
```
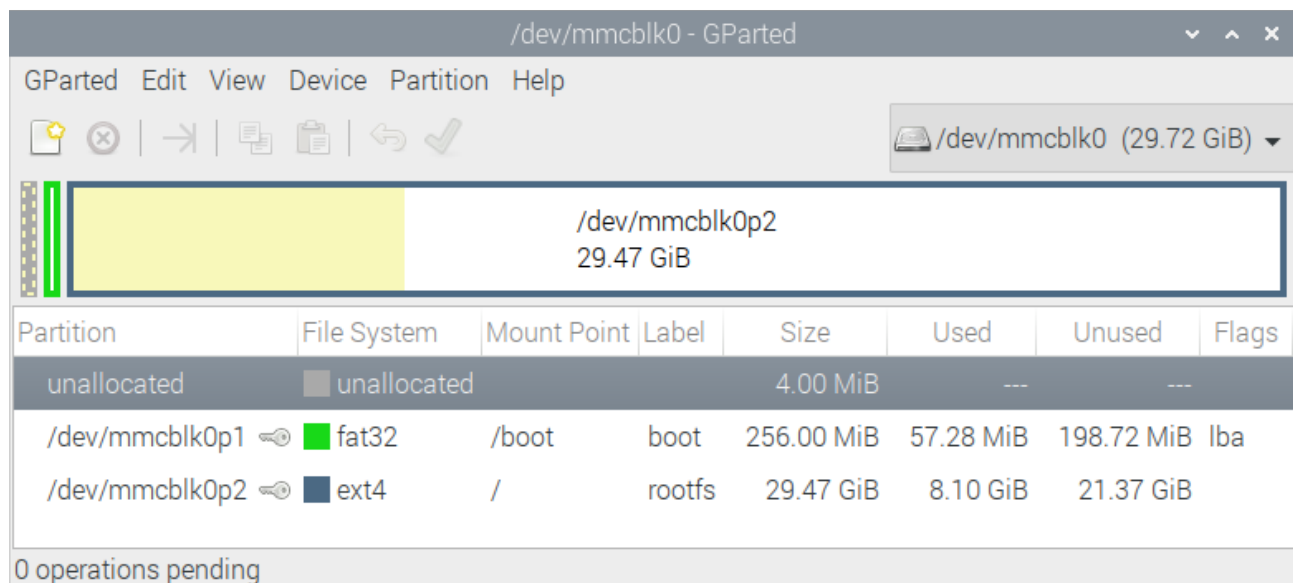
Gparted

1

Hardware

The RPI high camera system was installed on a Raspberry Pi 4 computer with 4 GB memory. The system will probably work on a Raspberry Pi 3 with 4 GB memory. 512 MB memory for GPU must be available if using a RPI with small amount (<4 GB) of memory.

Operating System

OS: Raspberry Pi OS
Linux raspberrypi 5.4.51-v7l+ #1333 SMP Mon Aug 10 16:51:40 BST 2020 armv7l GNU/Linux

cat /etc/os-release

PRETTY_NAME="Raspbian GNU/Linux 10 (buster)"
NAME="Raspbian GNU/Linux"
VERSION_ID="10"
VERSION="10 (buster)"
VERSION_CODENAME=buster
ID=raspbian
ID_LIKE=debian
HOME_URL="http://www.raspbian.org/"
SUPPORT_URL="http://www.raspbian.org/RaspbianForums"
BUG_REPORT_URL="http://www.raspbian.org/RaspbianBugs"

Software Versions

Python: 3.7.3
OpenCV: 4.4.0
rpi-camera suite: 15.9.2020

Memory Card Images

Clonezilla

Clonezilla version used to create this image: clonezilla-live-20180712-cosmic-amd64
URL: https://clonezilla.org
Image name: **rpi4-opencv4.4-2020-08-21-img**
Tar image name: **4-opencv4.4-2020-08-21-img.tar**

Clonezilla - Release 2020.08.26

Clonezilla version used to create this image: clonezilla-live-20180712-cosmic-amd64
URL: https://clonezilla.org
Image name: **rpi4-opencv4.4.0-scipy1.5.2-2020-08-26-img**
Tar image name: **rpi4-opencv4.4.0-scipy1.5.2-2020-08-26-img.tar**

Clonezilla - Release 2020.09.16 (current version)

Clonezilla version used to create this image: clonezilla-live-2.6.7-28-amd64
URL: https://clonezilla.org
Image name: **rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img**
Tar image name: **rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar**

dd and RaspberryPi Image Shrinkwrap

Version: 8.30
Image name: **rpi4-opencv4.4-2020-08-24.img**
Image was shrunk with shrinkwrap.sh (URL: https://github.com/mtyka/shrinkwrap). New size of partitions:

```
Device          Boot  Start      End  Sectors  Size Id Type
/dev/loop1000p1       8192   532479   524288  256M  c W95 FAT32 (LBA)
/dev/loop1000p2     532480 17920160 17387681  8.3G 83 Linux
```

Gzip image name: **rpi4-opencv4.4-2020-08-24.img.gz**

dd and RaspberryPi Image Shrinkwrap – Release 30.08.2020

Version: 8.30
Image name: **rpi4-opencv4.4.0-scipy1.5.2-2020-08-30.img**
Image was shrunk with shrinkwrap.sh. New size of partitions:

```
Device          Boot  Start      End  Sectors  Size Id Type
/dev/loop1000p1       8192   532479   524288  256M  c W95 FAT32 (LBA)
/dev/loop1000p2     532480 18154832 17622353  8,4G 83 Linux
```

Gzip image name: **rpi4-opencv4.4.0-scipy1.5.2-2020-08-30.img.gz**

Deploying a Clonezilla image to a micro-SD card

Download the Clonzilla archive and SHA256SUMS file from rpi-camera/images. Verify the SHA-256 hash:

Release 2020.08.21

```
sha256sum -c SHA256SUMS 2>&1 | grep OK
rpi4-opencv4.4-2020-08-21-img.tar: OK
```

Extract the directory and files using the following command:

```
tar -xvf rpi4-opencv4.4-2020-08-21-img.tar
```

Release 2020.08.26

```
sha256sum -c SHA256SUMS 2>&1 | grep OK
rpi4-opencv4.4.0-scipy1.5.2-2020-08-26-img.tar: OK
```

Extract the directory and files using the following command:

```
tar -xvf rpi4-opencv4.4.0-scipy1.5.2-2020-08-26-img.tar
```

Deploy the extracted image to a new ≥32 GB mico-SD card using Clonezilla.

Release 2020.09.16

```
$ sha256sum -c SHA256SUMS 2>&1 | grep OK
rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar-aa: OK
rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar-ab: OK
rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar-ac: OK
```

Recreate the image

```
$ cat rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar-* >
rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar
```

```
$ rm rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar-*
```

Download SHA252SUMS from https://github.com/kmiikki/rpi-camera/tree/master/images to image directory. Verify the SHA256 hash:

```
$ sha256sum -c SHA256SUMS 2>&1 | grep OK
rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar: OK
```

Extract the directory and files using the following command:

```
$ tar -xvf rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img.tar
```

=>

```
rpi4__opencv4.4-scipy1.5.2-roi-suite__2020-09-16-img/
```

Deploy the extracted image to a new ≥32 GB mico-SD card using Clonezilla.

Deploying a dd image to a micro-SD card

Download the compressed dd archive and SHA256SUMS file from rpi-camera/images. Verify the SHA-256 hash:

Release 2020.08.24

```
sha256sum -c SHA256SUMS 2>&1 | grep OK
rpi4-opencv4.4-2020-08-24.img.gz: OK
```

Extract the image file (size ~32 GB) from the compressed file:

```
gunzip -k rpi4-opencv4.4-2020-08-24.img.gz
```

NB The -k option keeps the input file.

Release 2020.08.30

```
sha256sum -c SHA256SUMS 2>&1 | grep OK
rpi4-opencv4.4.0-scipy1.5.2-2020-08-30.img.gz: OK
```

Extract the image file (size ~32 GB) from the compressed file:

```
gunzip -k rpi4-opencv4.4.0-scipy1.5.2-2020-08-30.img.gz
```

Get the micro-SD device name from lsblk output, i.e.:

```
lsblk
NAME           MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sde             8:64    1 14.8G  0 disk
├─sde1          8:65    1  256M  0 part
└─sde2          8:66    1 14.5G  0 part
```

Clone the image to a new ≥32 micro-SD card using following command:

```
sudo dd bs=4M if=rpi4-opencv4.4.0-scipy1.5.2-2020-08-30.img
of=/dev/sde conv=fsync status=progress
```

Clone the extracted image

NB It is possible to clone the shrunk dd image to a 16 GB micro-SD card. An error will be thrown (dd: error writing 'dev/…': No space left on device) which can be ignored.

Post Imaging Tasks

Insert the cloned micro-SD card into a Raspberry Pi. Start the computer and expand the root parition to fill SD card (if a shrinked dd image was used):

```
sudo raspi-config

7 Advanced Options <Enter>
A1 Expand Filesystem Ensures that all of the SD card storage is
available <Enter>
<Ok>
<Finish>
Reboot now? <Yes>
```

Cloning with dd and shrinkwrap image

This task is mandatory before anything else can be done! Doing something else (like installing programs) will probably make the system unusable.

Finnish keyboard layout can be changed following way:

Preferences > Raspberry Pi Configuration
Select Localisation tab, press <Set Keyboard...> and select form the Layout drop down list the desired keyboard layout.

Change the default password

Default user: pi
Default password: raspberry

Preferences > Raspberry Pi Configuration
Press <Change Password...>

Update the system and software

```
sudo apt update && sudo apt dist-upgrade
```

Install SciPy (< Release 2020.08.26)

```
$ sudo apt install python3-scipy
```

Update numpy, matplotlib, opencv-python and install spyder (Release ≤ 2020.08.30)

$ sudo apt install libatlas3-base libgfortran5
$ sudo apt install libgtk2.0-dev pkg-config

$ sudo pip3 install dlib face_recognition imutils
$ sudo pip3 install pysimplegui
$ sudo pip3 install Cython
$ sudo pip3 install scikit-build

$ sudo pip3 install numpy==1.19.1 --extra-index-url https://www.piwheels.org/simple
$ sudo pip3 install matplotlib==3.3.1 --extra-index-url https://www.piwheels.org/simple

Uninstall and build opencv-python

$ sudo pip3 uninstall opencv-python
$ sudo pip3 install --no-binary :all: opencv-python
Build time: ~2 h

$ sudo apt install spyder3

Copy ROI applications (roi*.py) to /opt/tools directory
$ sudo chmod ugo+rx /opt/tools/*.py