

Raspbian と ソフトウェアで広がる 情報工学の世界

金子邦彦

1. 前準備

① Raspbian のインストール

Raspbian のインストールは、次のページで説明している。
<https://www.kkaneko.jp/tools/raspbian/raspbian.html>

② Raspbian の初期設定と種々のソフトウェアのインストール

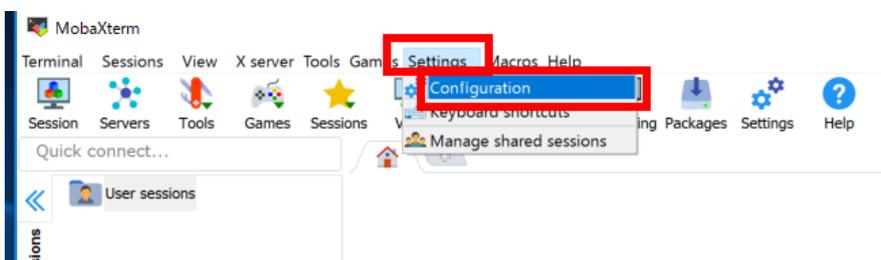
Raspbian の初期設定と種々のソフトウェアのインストールを、一度に簡単実行できる手順を次のページで説明している。
<https://www.kkaneko.jp/tools/raspbian/raspbiando.html>

2. Windows からラズベリーパイ(Raspberry Pi) にリモート接続

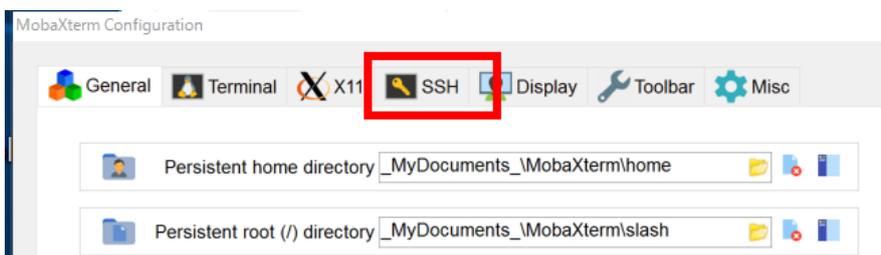
① MobaXTerm を起動

② MobaXTerm の設定で、「SSH keepalive」を有効にしておく（この設定は、一度だけ行っておけば大丈夫）

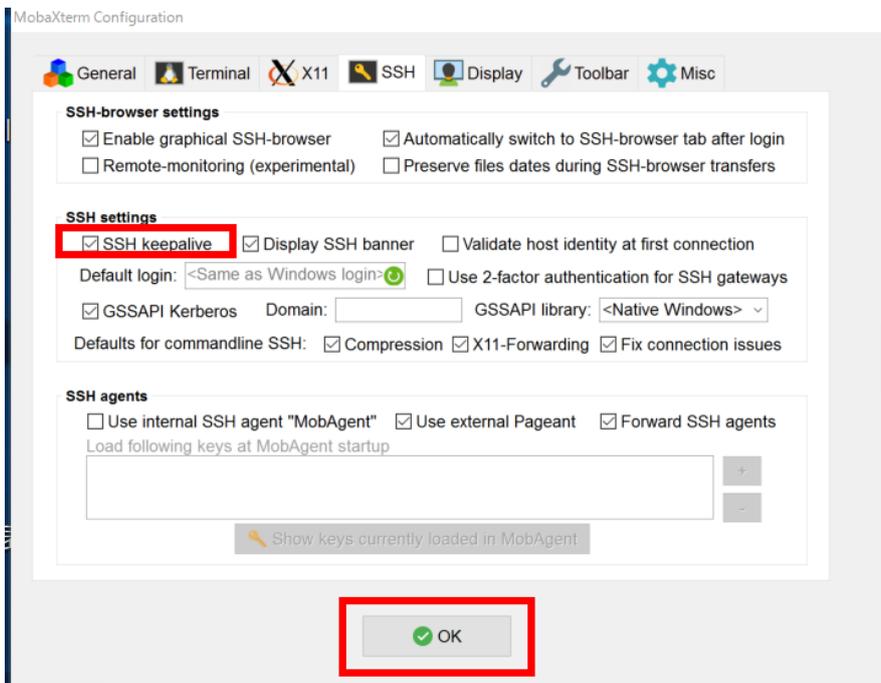
- ・ 「Settings」で「Configuration」を選ぶ



- ・ 「SSH」をクリック

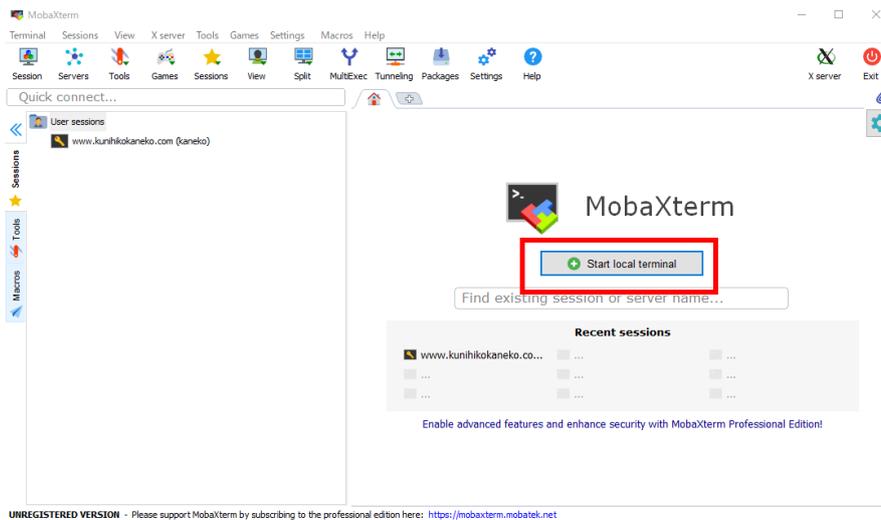


- ・ 「SSH keepalive」 をチェックし、「OK」 をクリック

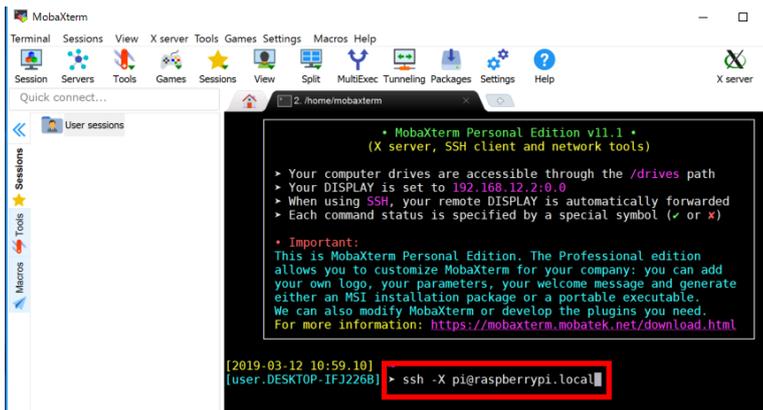


③ MobaXTerm を用いてリモートログイン

- ・ MobaXTerm で「Start local terminal」 をクリック



- ・ MobaXTerm のコンソールで「ssh -X pi@raspberrypi.local」 を実行
「pi@... password:」 に対しては「raspberry」、Enter キー
このとき、キーボードから入れたものが表示されないのは正常動作



・いま確認した IP アドレスを用いて、MobaXTerm のコンソールで、「ssh -X pi@<IP アドレス>」を実行して、リモートログインする

```
[2019-07-01 09:40:45] ~
[user.DESKTOP-IFJ226B] > ssh -X pi@192.168.12.3
pi@192.168.12.3's password:
```

「pi@... password:」に対しては「raspberry」、Enter キー
このとき、キーボードから入れたものが表示されないのは正常動作

```
[user.DESKTOP-IFJ226B] > ssh -X pi@192.168.12.3
pi@192.168.12.3's password:
Linux raspberrypi 4.19.50-v7+ #896 SMP Thu Jun 20 16:11:44 BST 2019 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jul 1 00:17:10 2019

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~ $
```

Windows パソコンから MobaXTerm へのリモートログインについては、次のページでも説明している
<https://www.kkaneko.jp/tools/raspbian/raspbian.html>

3. Linux コマンド

<用語集>

- **ファイル**

コンピュータで、データを保存するとき、ファイル名を付けて、ひとつのまとまりとして保存する。

- **ディレクトリ**

コンピュータでファイルを扱うとき、ファイルをグループ化して整理する。これをディレクトリという。

- **カレントディレクトリ**

コンピュータでファイルを扱うとき、現在、使用中のディレクトリのことをカレントディレクトリという

- **エディタ**

ファイルの編集機能をもったソフトウェアのことをエディタという。

<操作の例>

- **ファイル名表示**

カレントディレクトリ内の全ファイルのファイル名を表示

```
ls
```

```
pi@raspberrypi:~ $ ls
Desktop  Downloads  Music      Public     Videos
Documents  MagPi     Pictures   Templates
```

- **カレントディレクトリの変更**

カレントディレクトリを「/tmp」に変更

```
cd /tmp
```

```
pi@raspberrypi:~ $ cd /tmp
pi@raspberrypi:/tmp $ ls
dhcpcd-pi
ssh-Np0SKm2WcKt1
ssh-sHrvPhnxbHYp
ssh-tBUtUSkTSp
systemd-private-aa05fd71732843d791195a60e7af8f9c-systemd-timesyncd.service-L8bM
sj
pi@raspberrypi:/tmp $
```

- **ファイルの中身の表示**

ファイル名「hoge」のファイルの中身を表示

```
cat hoge
```

```
pi@raspberrypi:/tmp $ cat hoge
hello
pi@raspberrypi:/tmp $
```

<主な Linux コマンド>

ファイル操作

cat ファイル表示
ls ファイル名表示
cp ファイルコピー
mv ファイルの移動, ファイル名の変更
rm ファイル削除
cd カレントディレクトリの移動
pwd カレントディレクトリの表示
which コマンドのフルパス表示

システム

ps プロセス表示
df ディスク情報の表示
sudo 管理者権限でコマンドを実行
passwd パスワードの変更
shutdown システムのシャットダウン
uname -a カーネルのバージョンなどの表示

パッケージ

apt install パッケージのインストール
apt-get search パッケージの検索
dpkg -L インストール済みパッケージの中身の表示
dpkg -l インストール済みパッケージの表示

ネットワーク

ip neigh APR テーブルの表示
ip -s addr IP アドレスの表示
ip -s link ネットワークデバイスの使用状況
ping パケットを送り, 応答を調べる
ss ソケットの状態表示

ハードウェア

lsusb USB デバイスの一覧表示
lspci PCI デバイスの一覧表示
lshw ハードウェア情報の表示

4. 便利なアプリケーション

(1) nano (エディタ)

nano はエディタ

コマンド: **nano**

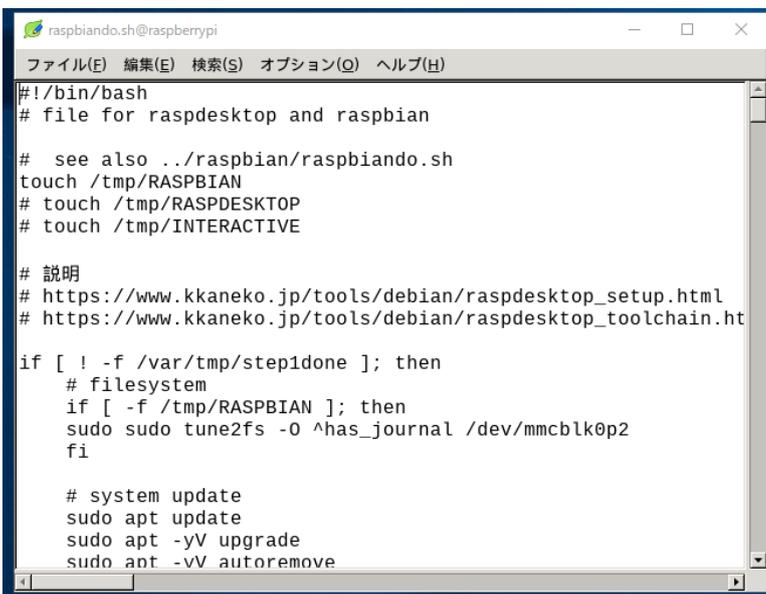


※ nano は、**CTRL (コントロールキー) , x の同時押しで終了**する。終了前に画面が切り替わる。
nano の終了時にファイルを保存したいときは **y** キー。そして、ファイル名を入れて **Enter** キー。

(2) leafpad (エディタ)

leafpad はエディタ

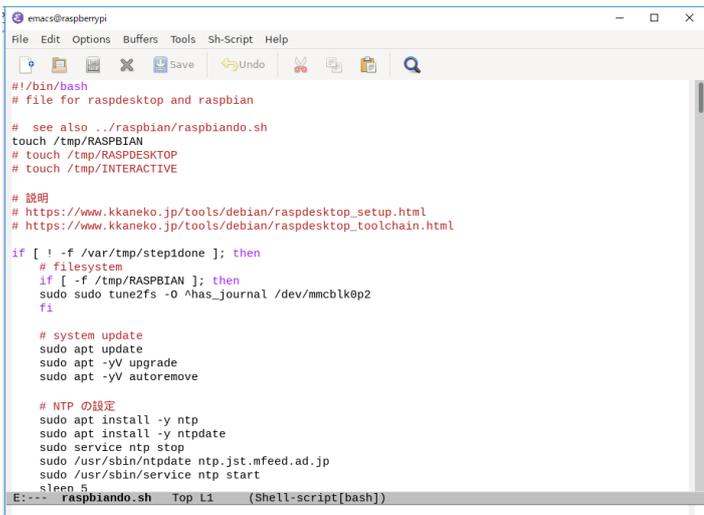
主要コマンド: **leafpad**



(3) emacs (エディタ)

emacs はエディタ

コマンド: **emacs**



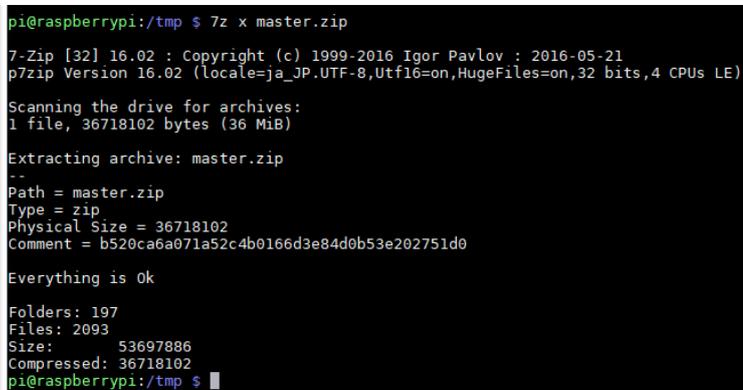
```
emacs@raspberrypi
File Edit Options Buffers Tools Sh-Script Help
# /bin/bash
# file for raspdesktop and raspbian
# see also ../raspbian/raspbiando.sh
touch /tmp/RASPBIAN
# touch /tmp/RASPDSTKTOP
# touch /tmp/INTERACTIVE
# 説明
# https://www.kkaneko.jp/tools/debian/raspdesktop_setup.html
# https://www.kkaneko.jp/tools/debian/raspdesktop_toolchain.html
if [ ! -f /var/tmp/step1done ]; then
# filesystem
if [ -f /tmp/RASPBIAN ]; then
sudo sudo tune2fs -o ^has_journal /dev/mmcblk0p2
fi
# system update
sudo apt update
sudo apt -y upgrade
sudo apt -y autoremove
# NTP の設定
sudo apt install -y ntp
sudo apt install -y ntpdate
sudo service ntp stop
sudo /usr/sbin/ntpdate ntp.jst.nfeed.ad.jp
sudo /usr/sbin/service ntp start
sleep 5
E: ... raspbiando.sh Top L1 (Shell-script[bash])
```

(4) p7zip

p7zip は UNIX(Linux, FreeBSD, Cygwin など) や MacOS で動くファイルアーカイバ。「7z x」で展開(解凍)

URL: <http://p7zip.sourceforge.net/>

コマンド: **7z**



```
pi@raspberrypi:/tmp $ 7z x master.zip
7-Zip [32] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 2016-05-21
p7zip Version 16.02 (locale=ja_JP.UTF-8,Utf16=on,HugeFiles=on,32 bits,4 CPUs LE)

Scanning the drive for archives:
1 file, 36718102 bytes (36 MiB)

Extracting archive: master.zip
--
Path = master.zip
Type = zip
Physical Size = 36718102
Comment = b520ca6a071a52c4b0166d3e84d0b53e202751d0

Everything is Ok

Folders: 197
Files: 2093
Size: 53697886
Compressed: 36718102
pi@raspberrypi:/tmp $
```

(5) Blender (3次元)

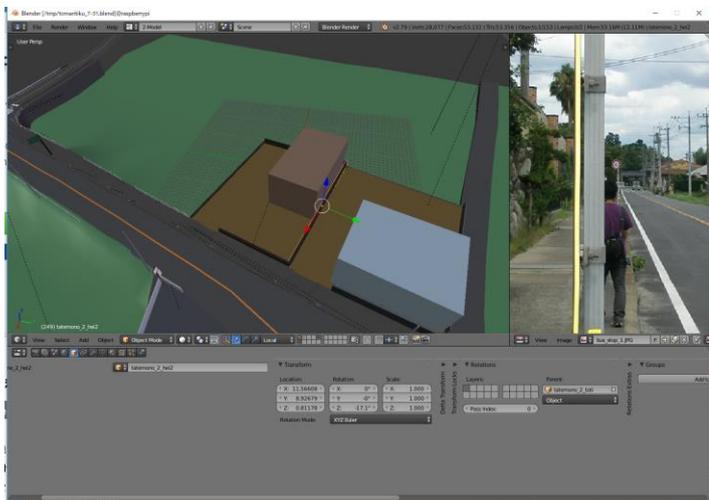
Blender は 3次元のソフトウェア。モデリング, リギング, アニメーション, シミュレーション, レンダリングの機能がある。

URL: <https://www.blender.org/>

ライセンス: GNU General Public License

コマンド: **blender** (起動)

```
wget https://www.kkaneko.jp/data/3ddata/tomaritiku_7-31.blend
blender tomaritiku_7-31.blend
```



使用法は次のページで説明している.

<https://www.kkaneko.jp/dblab/cg/blenderintro.html>

(6) MakeHuman (人体アニメーション)

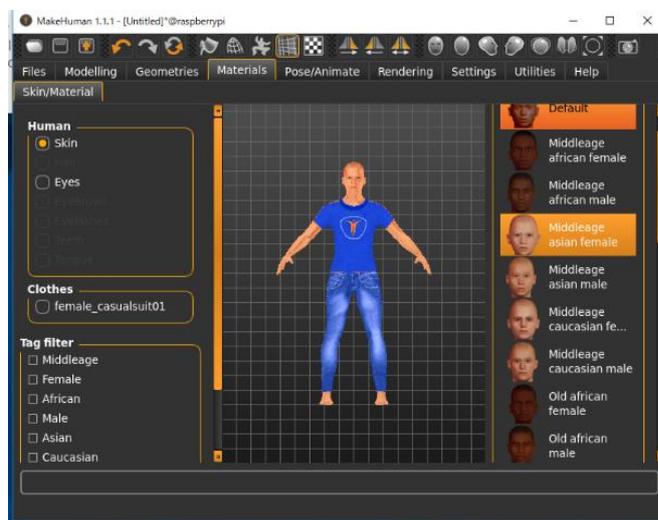
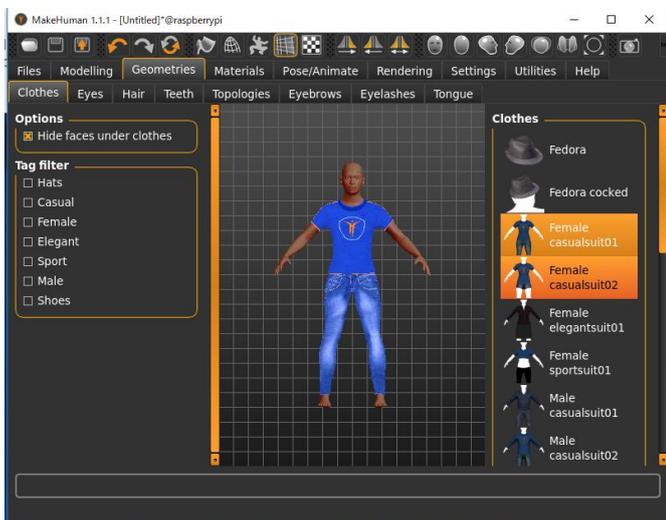
MakeHuman は人体アニメーションのソフトウェア

URL: <http://www.makehumancommunity.org/>

ライセンス : GNU Affero General Public License,

<http://www.makehumancommunity.org/content/license.html>

コマンド: **makehuman** (起動)



使用法は次のページで説明している.

<https://www.kkaneko.jp/dblab/cg/makehuman.html>

(7) MeshLab (3次元)

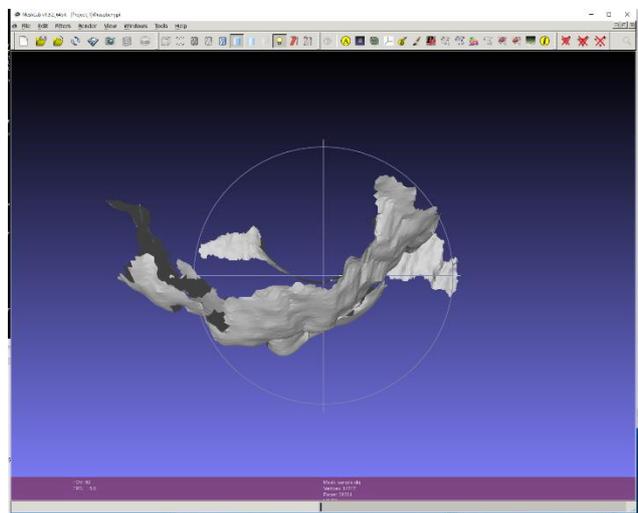
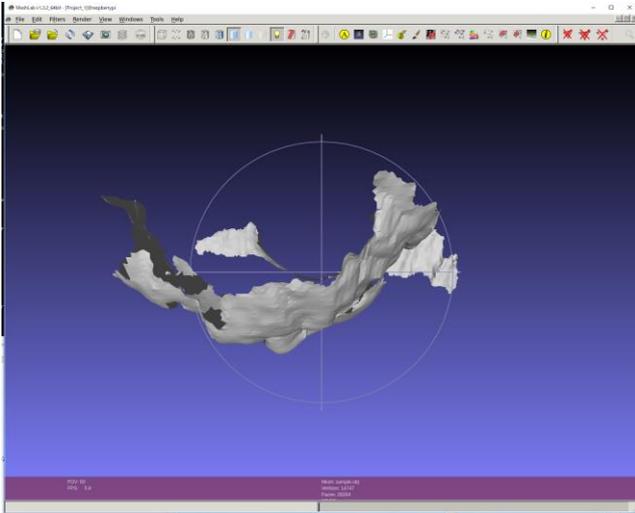
MeshLab は, 3次元メッシュの編集と処理のソフトウェア.

URL: <http://www.meshlab.net/>

ライセンス: GNU General Public License

主要コマンド: **meshlab** (起動)

```
wget https://www.kkaneko.jp/data/3ddata/sample.obj  
wget https://www.kkaneko.jp/data/3ddata/sample.mtl  
meshlab sample.obj
```



使用法は次のページで説明している.

<https://www.kkaneko.jp/dblab/pointcloud/meshlabpc.html>

(8) ImageMagick (画像)

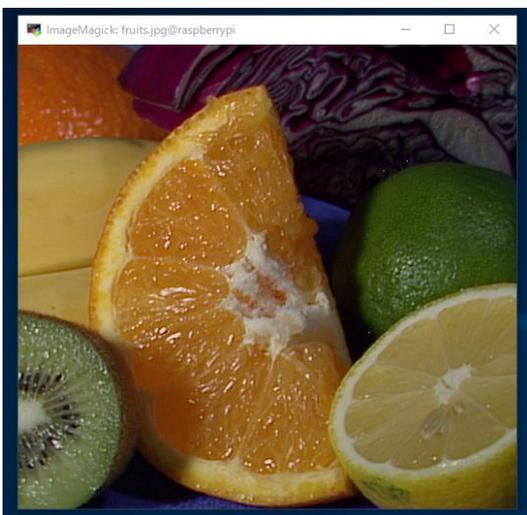
ImageMagick は、画像の生成、編集、合成、返還の機能を持ったソフトウェア.

URL: <https://imagemagick.org/index.php>

ライセンス: derived Apache 2.0 license

主要コマンド: **magick**, **display**

```
display /usr/local/share/opencv4/samples/data/fruits.jpg
```



(9) ffmpeg (ビデオデータ処理)

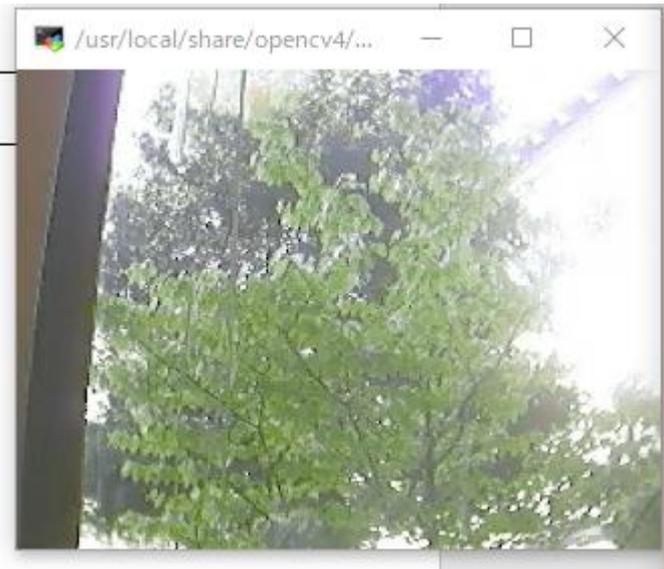
ffmpeg は、動画の変換、再生、解析の機能を持ったソフトウェア

URL: <https://ffmpeg.org/>

ライセンス: LGPL version 2.1, GPL version 2 (例外規定あり)

コマンド: **ffmpeg**, **ffmpeg-all**, **ffplay**, **ffplay-all**, **ffprobe**, **ffprobe-all**

```
ffplay /usr/local/share/opencv4/samples/data/tree.avi
```



(10) OpenShot (動画)

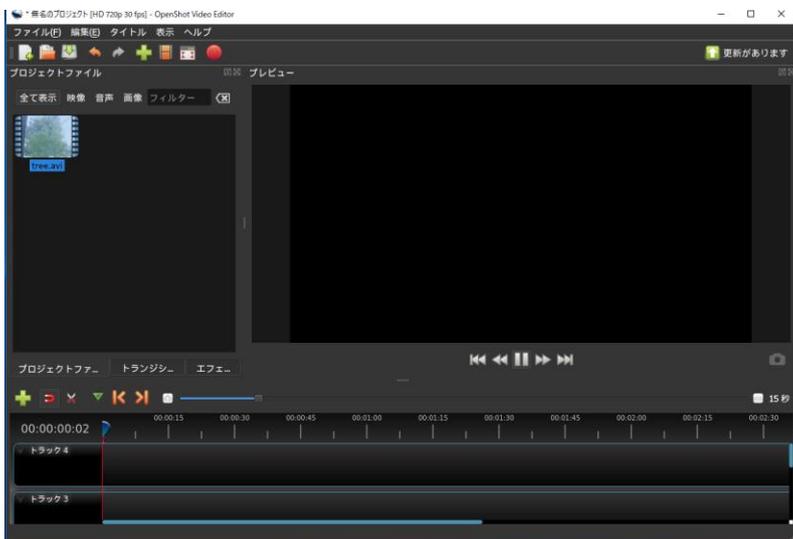
OpenShot は動画編集ソフトウェア

URL: <https://www.openshot.org/ja/>

ライセンス: GNU General Public License 3

コマンド: **openshot-qt**

```
openshot-qt
```



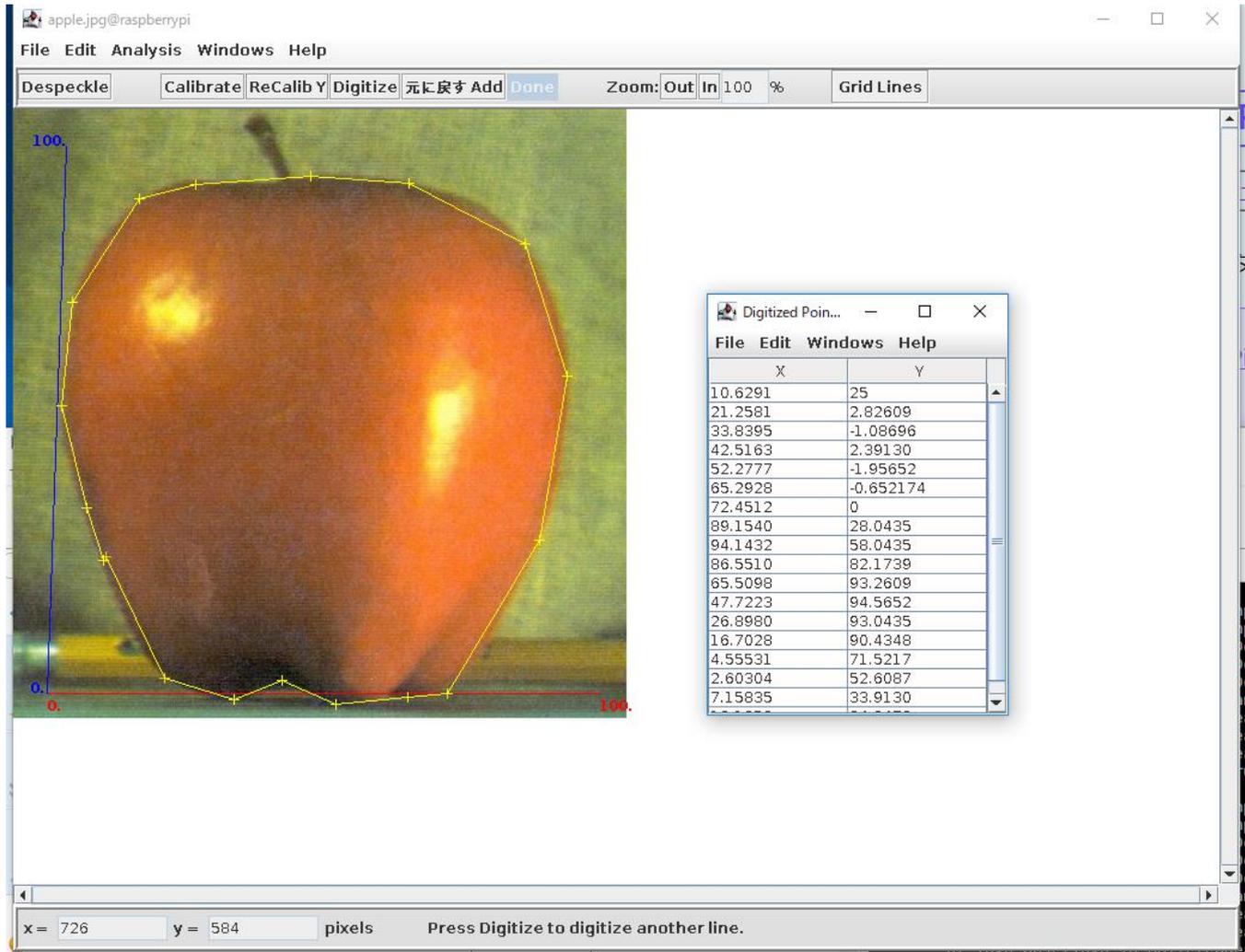
(11) Plot Digitizer (デジタイザ)

Plot Digitizer はデータポイントをデジタイザする機能をもったソフトウェア

URL: <https://sourceforge.net/projects/plotdigitizer/>

ライセンス: GNU Library or Lesser General Public License version 2.0 (LGPLv2), GNU General Public License version 2.0 (GPLv2)

コマンド: `java -jar /usr/local/PlotDigitizer_Linux_and_Others/PlotDigitizer.jar`



(12) Tesseract OCR (デジタイザ)

Tesseract OCR は、文字認識のソフトウェア

URL: <https://sourceforge.net/projects/plotdigitizer/> <https://github.com/tesseract-ocr/tesseract>

ライセンス: Apache License v2.0

コマンド: `tesseract`

```
tesseract <画像ファイル名> outbase l jpn
```

```
pi@raspberrypi:/tmp $ tesseract 30.png outbase -l jpn
Tesseract Open Source OCR Engine v4.0.0 with Leptonica
pi@raspberrypi:/tmp $ more outbase.txt
Raspbian とソフトウェアで広がる。
情報工学の世界。

2019年 7月。

1. Windows からラズベリーパイ (Raspberry Pi) にリモート接続。

@-MobaXTerm を起動
@MobaXTerm の設定で、「SSH-keepalive」を有効にしておく（この設定は、一度だけ行っておけば大丈夫）。

↑
pi@raspberrypi:/tmp $
```

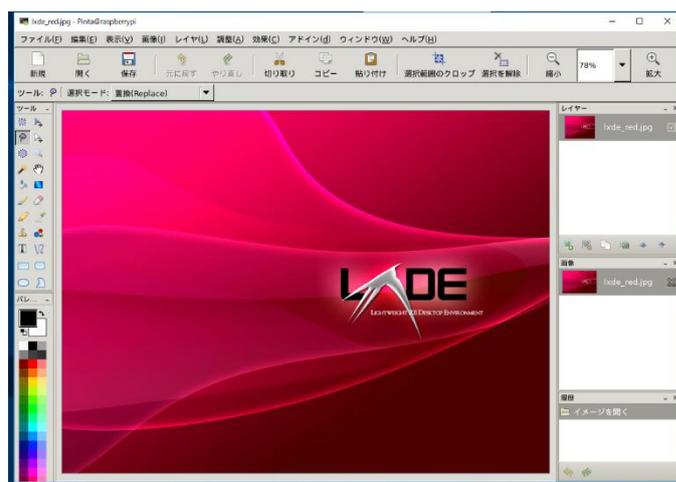
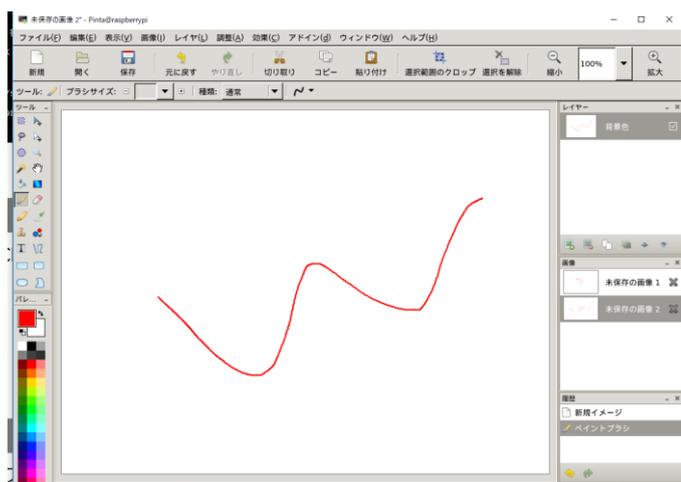
(13) Pinta (画像編集)

Pinta は、画像描画、画像編集のソフトウェア

URL: <https://pinta-project.com/pintaproject/pinta/>

ライセンス: MIT License

コマンド: **pinta**



(14) wget (ファイル検索)

wget は HTTP, HTTPS, FTP, FTPS によるファイル検索ツール

URL: <https://www.gnu.org/software/wget/>

ライセンス: GNU General Public License (例外規定あり)

コマンド: **wget**

```
pi@raspberrypi:/tmp $ wget https://bootstrap.pypa.io/get-pip.py
--2019-06-30 09:19:09-- https://bootstrap.pypa.io/get-pip.py
bootstrap.pypa.io (bootstrap.pypa.io) を DNS に問い合わせています ... 2a04:4e42:15:
:175, 151.101.88.175
bootstrap.pypa.io (bootstrap.pypa.io)[2a04:4e42:15::175]:443 に接続しています ...
接続しました。
HTTP による接続要求を送信しました、応答を待っています ... 200 OK
長さ: 1709825 (1.6M) [text/x-python]
`get-pip.py' に保存中

get-pip.py 100%[=====>] 1.63M 8.91MB/s 時間 0.2s
2019-06-30 09:19:09 (8.91 MB/s) - `get-pip.py' へ保存完了 [1709825/1709825]
pi@raspberrypi:/tmp $
```

(15) Netcat (通信)

Netcat は、TCP や UDP の読み書き、TCP サーバ等の種々の機能を持ったソフトウェア

URL: <http://man.openbsd.org/nc>

コマンド: **nc**

(16) Wireshark (通信)

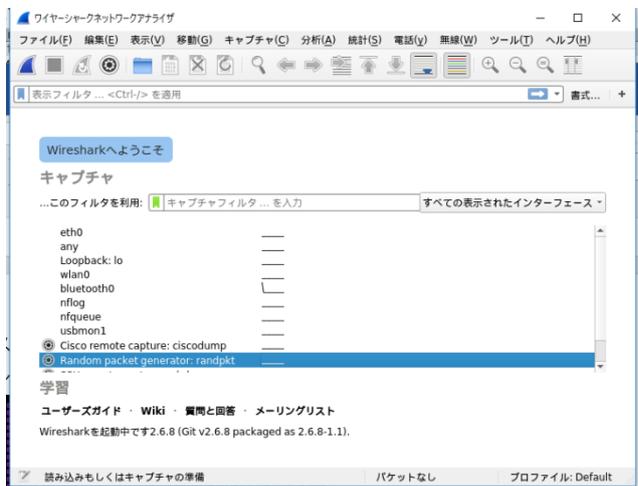
Wireshark は、ネットワークプロトコルアナライザ

URL: <https://www.wireshark.org/>

ライセンス: GNU General Public License

コマンド: **tshark** (キャプチャ), **wireshark** (解析)

```
Capturing on 'eth0'
  0.000000000 192.168.2.103 -> 230.0.0.1 UDP 92 49662 - 6666 Len=50
  2.0.48234875 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
  3.0.523524192 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=53 Win=2048 Len=0
  4.1.080337891 192.168.2.103 -> 230.0.0.1 UDP 92 49662 - 6666 Len=50
  5.1.012174928 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
  6.1.012476072 192.168.2.109 -> 192.168.2.103 SSH 122 Server: Encrypted packet (len=68)
  7.1.012759813 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=173 Win=2047 Len=0
  8.1.522598682 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
  9.1.522972347 192.168.2.109 -> 192.168.2.103 SSH 122 Server: Encrypted packet (len=68)
 10.1.523205367 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
 11.1.523293752 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=293 Win=2053 Len=0
 12.1.563713437 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=345 Win=2052 Len=0
 13.2.000681302 192.168.2.103 -> 230.0.0.1 UDP 92 49662 - 6666 Len=50
 14.2.052649173 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
 15.2.052740270 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
 16.2.052656879 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=449 Win=2052 Len=0
 17.2.052743181 192.168.2.109 -> 192.168.2.103 SSH 138 Server: Encrypted packet (len=84)
 18.2.052825688 192.168.2.109 -> 192.168.2.103 SSH 90 Server: Encrypted packet (len=36)
 19.2.053228596 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=509 Win=2052 Len=0
 20.2.562511854 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
 21.2.562832686 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
 22.2.563129195 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=673 Win=2051 Len=0
 23.2.563267059 192.168.2.109 -> 192.168.2.103 SSH 122 Server: Encrypted packet (len=68)
 24.2.563547735 192.168.2.109 -> 192.168.2.103 SSH 154 Server: Encrypted packet (len=100)
 25.2.563791589 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=841 Win=2050 Len=0
 26.2.708398768 192.168.2.102 -> 224.0.0.251 MDNS 154 Standard query 0x0000 PTR_companion-Link_tcp.local, *QM* question PTR_homekit_tcp.local, *QM* question PTR_sleep-proxy_udp.local, *QM* question OP
 27.2.708816206 fe80::1406:5535:Seca:54b4 -> ff02::fb MDNS 174 Standard query 0x0000 PTR_companion-Link_tcp.local, *QM* question PTR_homekit_tcp.local, *QM* question PTR_sleep-proxy_udp.local, *QM* question OP
 28.3.001017213 192.168.2.103 -> 230.0.0.1 UDP 92 49662 - 6666 Len=50
 29.3.092837637 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
 30.3.092854719 192.168.2.109 -> 192.168.2.103 SSH 106 Server: Encrypted packet (len=52)
 31.3.092872634 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=945 Win=2050 Len=0
 32.3.092835342 192.168.2.109 -> 192.168.2.103 SSH 138 Server: Encrypted packet (len=84)
 33.3.093257528 192.168.2.109 -> 192.168.2.103 SSH 234 Server: Encrypted packet (len=180)
 34.3.093488073 192.168.2.109 -> 192.168.2.103 SSH 122 Server: Encrypted packet (len=68)
 35.3.093556121 192.168.2.103 -> 192.168.2.109 TCP 60 18053 - 22 [ACK] Seq=1 Ack=1209 Win=2049 Len=0
```



(17) SQLite3 (データベース)

SQLite3 は軽量のリレーショナルデータベース管理システム

URL: <https://www.sqlite.org/index.html>

ライセンス: パブリックドメイン

コマンド: **sqlite3**

```
pi@raspberrypi:/tmp $ sqlite3 mydb
SQLite version 3.27.2 2019-02-25 16:06:06
Enter ".help" for usage hints.
sqlite> create table T (
...>   id integer primary key,
...>   name text );
sqlite> insert into T values(1, 'hoge');
sqlite> select * from T;
1|hoge
sqlite> .exit
pi@raspberrypi:/tmp $
```

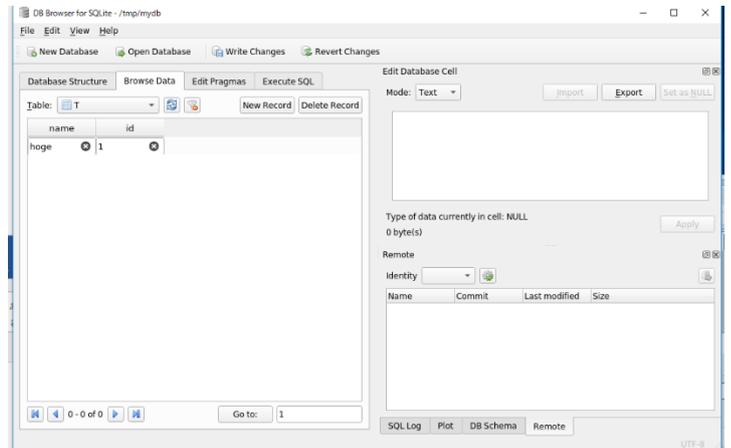
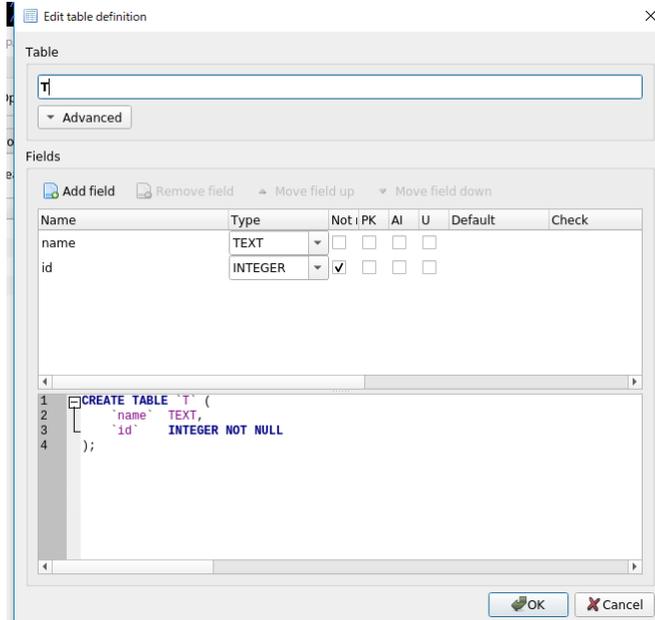
(18) DB Browser for SQLite (データベース)

DB Browser for SQLite は、SQLite データベースの作成、設計、編集の機能を持ったソフトウェア

URL: <https://sqlitebrowser.org/>

ライセンス: Mozilla Public License Version 2, GNU General Public License Version 3 or later

コマンド: **sqlitebrowser**



(19) hwinfo (ハードウェア情報)

hwinfo はハードウェア情報を表示する機能をもったソフトウェア

URL: [git://git.opensuse.org/projects/hwinfo.git](https://git.opensuse.org/projects/hwinfo.git)

コマンド: **hwinfo**

```
pi@raspberrypi:/tmp $ hwinfo --short
cpu:
    CPU
    CPU
    CPU
    CPU
network:
 wlan0      Broadcom BCM43430 WLAN card
 eth0      Standard Microsystems SMC9512/9514 Fast Ethernet Adapter
network interface:
 eth0      Ethernet network interface
 lo        Loopback network interface
 wlan0     WLAN network interface
disk:
 /dev/ram11  Disk
 /dev/ram2   Disk
 /dev/ram0   Disk
 /dev/ram9   Disk
 /dev/ram7   Disk
 /dev/ram14  Disk
 /dev/ram5   Disk
 /dev/ram12  Disk
 /dev/ram3   Disk
 /dev/ram10  Disk
 /dev/ram1   Disk
 /dev/ram8   Disk
 /dev/ram15  Disk
 /dev/ram6   Disk
 /dev/mmcblk0  Disk
 /dev/ram13  Disk
 /dev/ram4   Disk
partition:
 /dev/mmcblk0p1  Partition
 /dev/mmcblk0p2  Partition
usb controller:
    ARM USB controller
hub:
    Linux Foundation 2.0 root hub
    Standard Microsystems SMC9514 Hub
memory:
    Main Memory
unknown:
    Broadcom BCM43430 WLAN card
pi@raspberrypi:/tmp $
```

4. プログラム作成のためのアプリケーション類

(1) Python バージョン 3

URL: <https://www.python.org>

ライセンス: Python PSF LICENSE AGREEMENT

コマンド: **python3** (Python 言語処理), **pip3** (パッケージ管理)

```
pi@raspberrypi:~ $ python3
Python 3.7.3 (default, Apr 3 2019, 05:39:12)
[GCC 8.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print(1 + 2)
3
>>> exit()
pi@raspberrypi:~ $
```

Python はプログラミング言語.

- python3 起動.
- exit() 終了

```
pi@raspberrypi:~ $ pip3 show numpy
Name: numpy
Version: 1.16.2
Summary: NumPy is the fundamental package for array computing with Python.
Home-page: https://www.numpy.org
Author: Travis E. Oliphant et al.
Author-email: None
License: BSD
Location: /usr/lib/python3/dist-packages
Requires:
Required-by: sense-hat, scrollphathd, microdotphat
pi@raspberrypi:~ $
```

pip3 はパッケージ管理.

- sudo pip3 install パッケージのインストール
- pip3 list インストール済みのパッケージ一覧
- pip3 show 情報表示

(2) virtualenv, virtualenv wrapper

- mkvirtualenv <Python 仮想環境名> : Python 仮想環境の新規作成
- workon: Python 仮想環境の一覧表示
- workon <Python 仮想環境名> : Python 仮想環境の有効化
- deactivate: いま有効化されている Python 仮想環境の無効化

(3) TensorFlow バージョン 1.13.1, Keras バージョン 2.2.4

TensorFlow, Keras は、機械学習（ディープラーニングなど）のソフトウェア

URL: <https://www.python.org>

ライセンス: Apache License 2.0

```
>>> print(m.summary())
Layer (type)                Output Shape         Param #
-----
dense_1 (Dense)             (None, 64)          320
activation_1 (Activation)    (None, 64)          0
dense_2 (Dense)             (None, 3)           195
activation_2 (Activation)    (None, 3)           0
-----
Total params: 515
Trainable params: 515
Non-trainable params: 0
None
>>>
```

```
>>> m.predict( np.array([[0, 1, 0, 1]]) )
array([[0.15410425, 0.8341774 , 0.0117184 ]], dtype=float32)
>>>
```

(4) Caffe バージョン 1.0.0 (ディープラーニング)

Caffe はディープラーニングのソフトウェア。

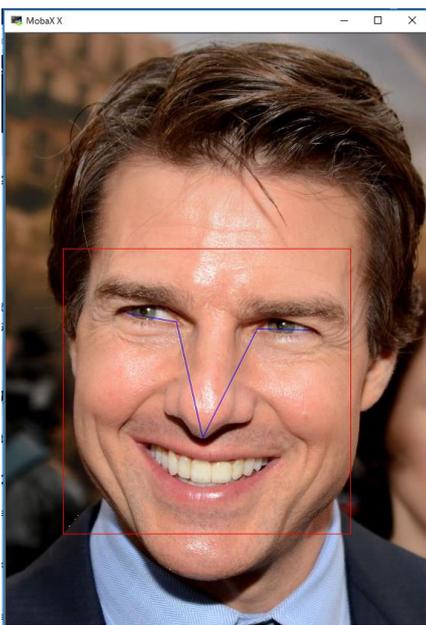
(5) Chainer バージョン 7.0.0b (ディープラーニング)

Chainer は Python で動くニューラルネットワークのソフトウェア。

(6) Dlib バージョン 19 (顔検知, 顔認識)

Dlib は, 顔検知, 顔認識のソフトウェア

```
python3 ../python_examples/face_recognition.py /usr/local/dlib/shape_predictor_5_face_landmarks.dat /usr/local/dlib/dlib_face_recognition_resnet_model_v1.dat
```

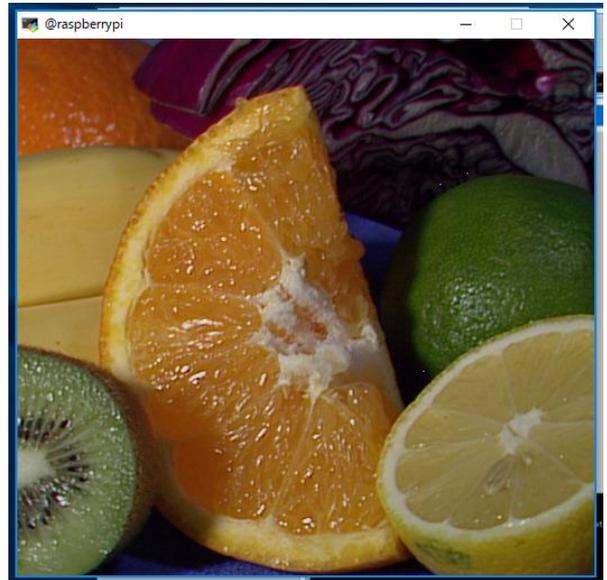


(7) OpenCV バージョン 4.1.1 pre (コンピュータビジョン)

OpenCV は, コンピュータビジョンのソフトウェア

```
import cv2
IMROOT="/usr/local/share/opencv4/samples/data/"
bgr = cv2.imread(IMROOT + "fruits.jpg")
cv2.imshow("", bgr)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
pi@raspberrypi:/tmp $ python3
Python 3.7.3 (default, Apr 3 2019, 05:39:12)
[GCC 8.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> IMROOT="/usr/local/image/"
>>> bgr = cv2.imread(IMROOT + "fruits.jpg")
>>> cv2.imshow("", bgr)
>>> cv2.waitKey(0)
```



(8) OpenALPR バージョン 2.3.0 (ナンバープレート)

OpenALPR はナンバープレート読み取りのソフトウェア

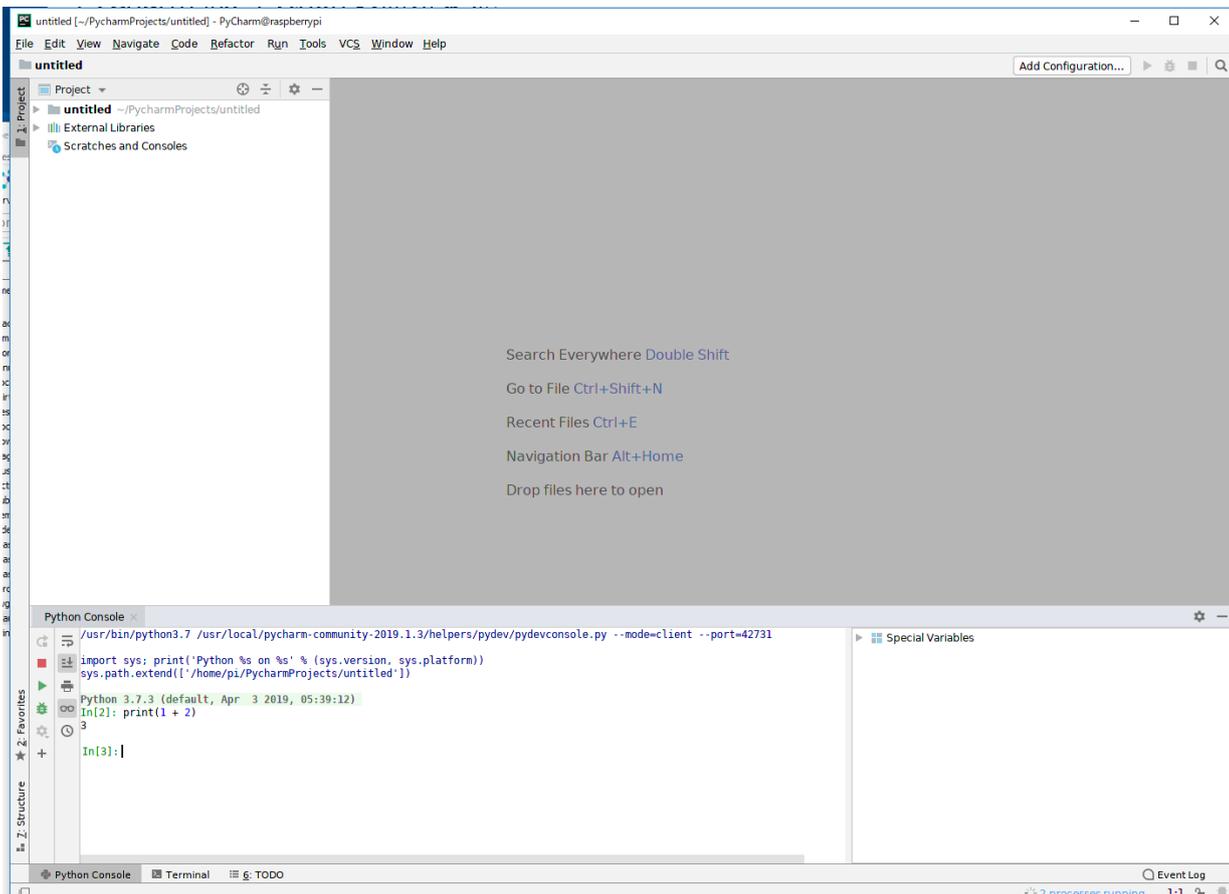
(9) PyCharm Community (Python 開発環境)

PyCharm は, Python の開発環境

URL: <https://www.jetbrains.com/pycharm/>

ライセンス: Apache 2 license

コマンド: **pycharm.sh** (PyCharm), **inspect.sh** (インスペクタ), **format.sh**



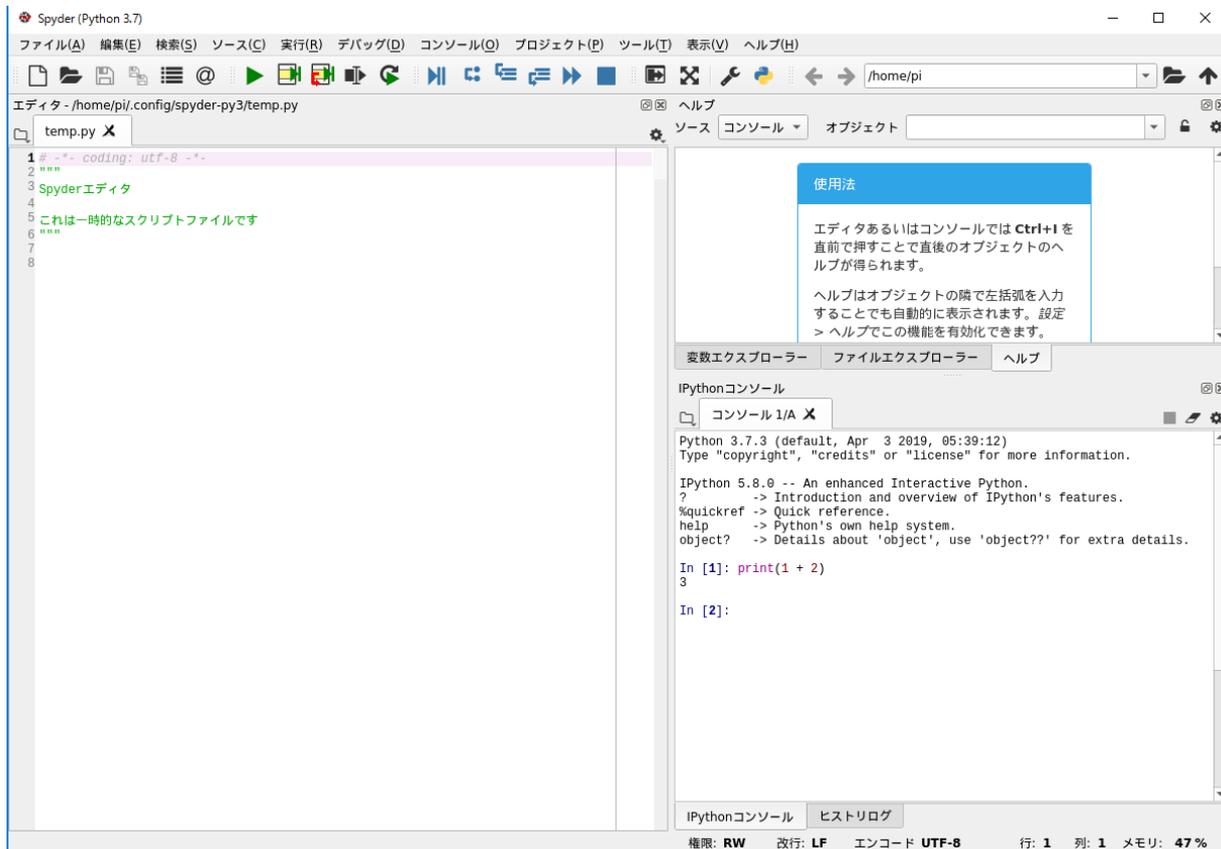
(10) spyder (Python 開発環境)

spyder は, Python の開発環境

URL: <https://github.com/spyder-ide/spyder/wiki>

ライセンス: MIT license

コマンド: **spyder3** (起動)



(11) BlueJ (Java 開発環境)

BlueJ は, Java の開発環境

URL: <https://www.bluej.org/>

ライセンス: GNU GENERAL PUBLIC LICENSE Version 2

主要コマンド: **bluej** (起動)

(12) git

URL: <https://git-scm.com/>

ライセンス: GNU General Public License version 2.0

主要コマンド: **git**

```
pi@raspberrypi:/tmp $ git clone https://github.com/tensorflow/tensorflow.git
Cloning into 'tensorflow'...
remote: Enumerating objects: 619420, done.
remote: Total 619420 (delta 0), reused 0 (delta 0), pack-reused 619420
Receiving objects: 100% (619420/619420), 351.84 MiB | 2.42 MiB/s, done.
Resolving deltas: 100% (501336/501336), done.
Checking out files: 100% (17628/17628), done.
pi@raspberrypi:/tmp $
```

git はバージョン管理ソフトウェア

- git clone レポジトリのクローンを行う

(13) cmake

URL: <https://cmake.org/>

ライセンス: OSI-approved BSD 3-clause License

主要コマンド: **cmake**

```
pi@raspberrypi:/tmp/dlib/build $ cmake ..
```

cmake はソフトウェアのビルド, テスト, パッケージ化の機能を持ったソフトウェア

```
pi@raspberrypi:/tmp/dlib/build $ cmake --build .
```

- cmake cmake による設定ファイルの生成

- cmake --build . ビルド

(14) GNU ツールチェーン

URL: <https://www.gnu.org/>

ライセンス: GNU General Public License (例外規定あり)

主要コマンド: **gcc, g++, gfortran, make, autoconf, automake, bison**