

Raspberry Pi の初期設定

目次

1	Raspberry Pi の初期設定	2
	System 設定.....	3
	Interfaces 設定	4
	Localisation 設定	5
2	無線 LAN の設定	7
	GUI での設定	7
	CUI での設定.....	7
3	PC の設定.....	11
	Bonjour のインストール (Windows のみ).....	11
	Bonjour がうまく機能しない場合	11
	TeraTerm のインストール	11
4	Raspberry Pi に OpenRTM-aist をインストール.....	13
	OpenRTM-aist のインストール.....	13
5	CMake のインストール	14
6	Doxygen のインストール.....	14
7	Subversion/Git のインストール	14
8	OpenCV のインストール.....	14

当ドキュメントは下記ページを参考にしています。

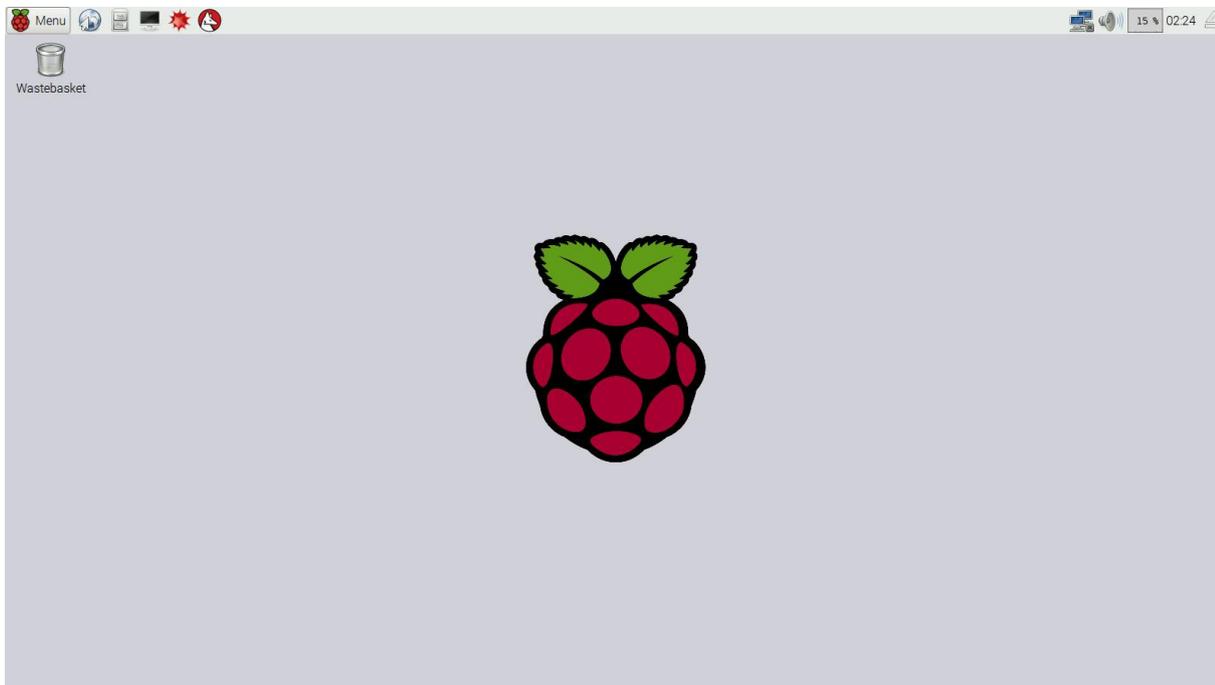
- Raspberry Pi の初期設定

http://www.openrtm.org/openrtm/ja/content/raspberrypi_initial_setting (2016/1/20 アクセス)

1 Raspberry Pi の初期設定

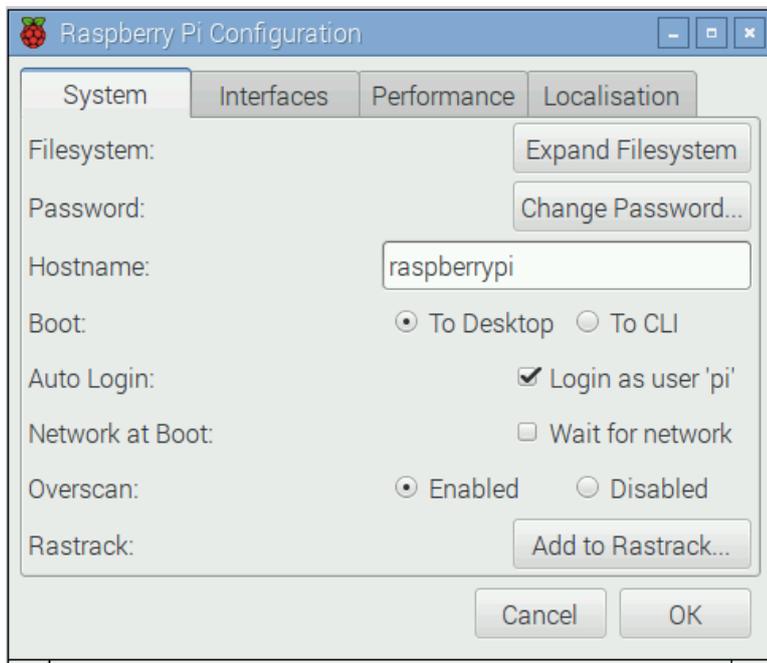
Raspberry Pi に HDMI モニタ、キーボード、マウスを接続してください。

SD カードを挿入し Raspberry Pi に始めて電源を投入すると、各種ドライバの読み込み画面が表示された後、以下の画面が表示されます。



System 設定

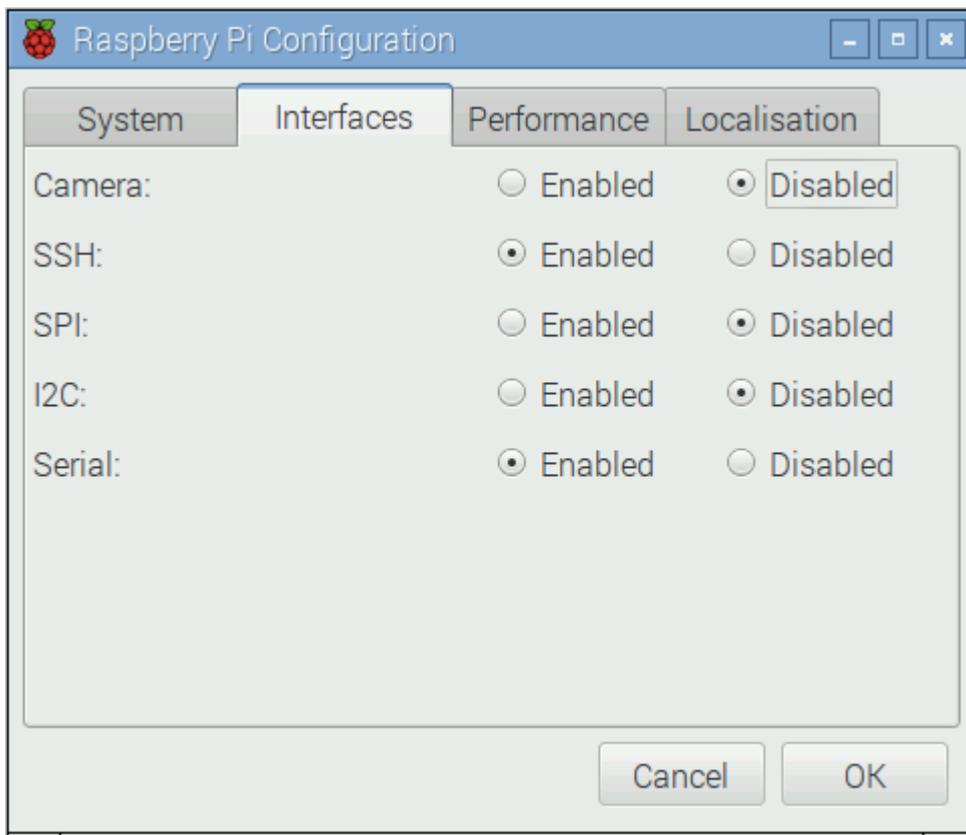
「Menu」 → 「Preferences」 → 「Raspberry Pi Configuration」 を選択すると以下の画面が表示されますので設定を行います。



- Filesystem : 「Expand Filesystem」 ボタンをクリックします。
- Hostname : 他のホスト名と衝突しないホスト名を入力します。
- Boot : 「To CLI」 を選択します。
- Auto Login : 「Login as user 'pi」 のチェックをはずします。

Interfaces 設定

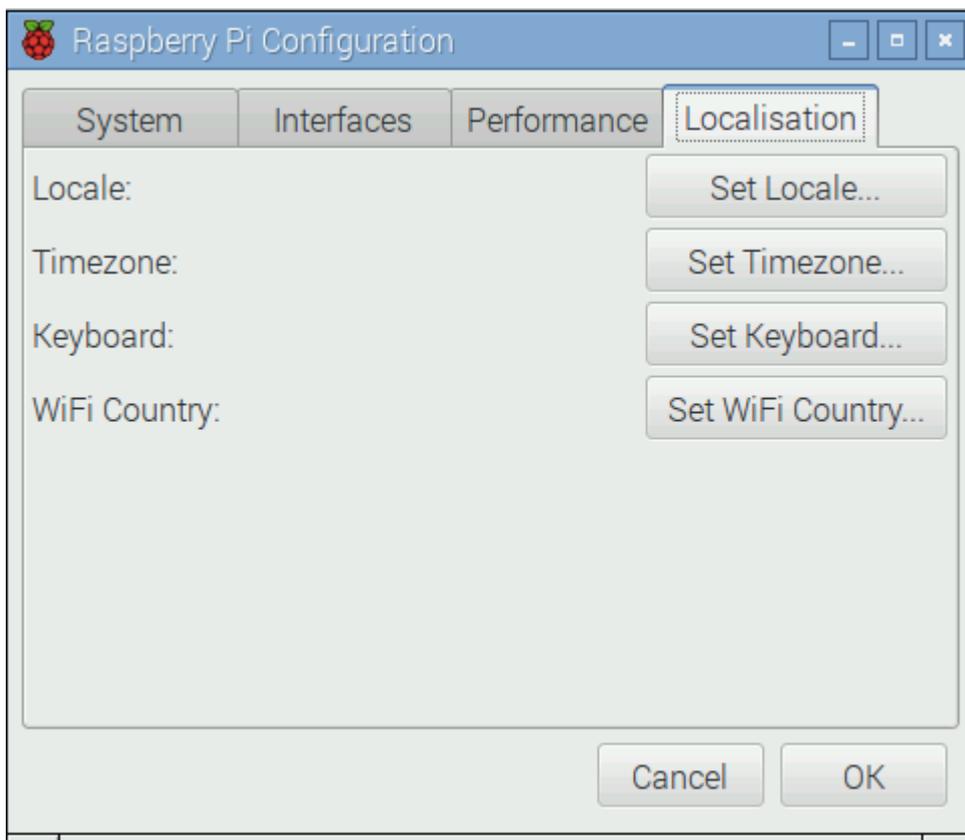
「Interfaces」タブをクリックすると以下の画面が表示されますので設定を行います。



- Camera : 「Enabled」を選択します。
- SPI : 「Enabled」を選択します。
- I2C : 「Enabled」を選択します。

Localisation 設定

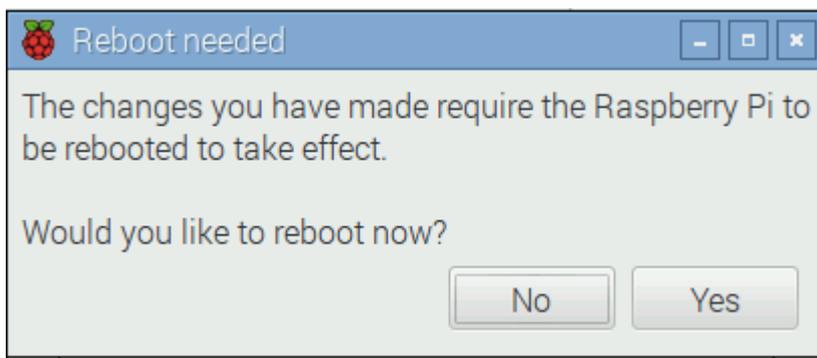
「Localisation」タブをクリックすると以下の画面が表示されますので設定を行います。



- Keyboard : 「Set Keyboard」 → 「Japan (Country)」 → 「Japanese (Variant)」

設定が終わりましたら「OK」ボタンをクリックします。

以下の再起動確認画面が表示されますので「YES」を選択します。



再起動しましたら、以下の ID、パスワードでログインします。

ID : pi

パスワード: raspberry

終了する場合は、以下のコマンドで停止した後、本体から電源を抜いてください。

```
$ sudo halt
```

sudo : 指定したユーザーでコマンドを実行する。デフォルトでは **root** 権限で実行する。

halt : システムをシャットダウンする。

2 無線 LAN の設定

GUI での設定

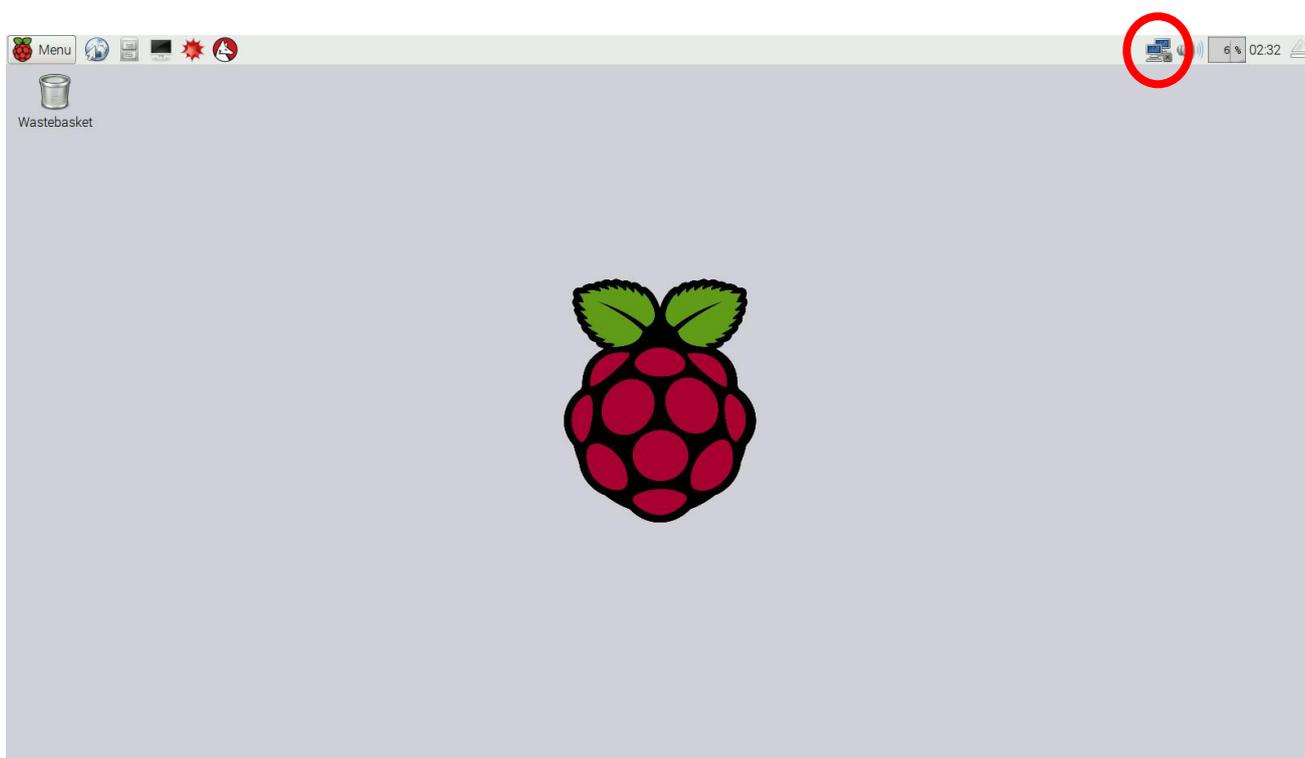
接続したい無線 LAN に接続可能な場所で設定を行う場合は GUI での設定が可能です。

※GUI で設定を行う場合にはパスキーの暗号化がされません。パスキーの暗号化を行いたい場合には CUI での設定を行ってください。

ログイン後に以下のコマンドを入力します。

```
$ startx
```

startx : GUI モードを起動する。



画面右上のコンピュータが2つ並んだアイコンをクリックすると接続可能なワイヤレスネットワーク一覧が表示されますので目的のワイヤレスネットワークを選択し、「Pre Shared Key」にパスフレーズを入力し「OK」ボタンをクリックしますと接続されます。

CUI での設定

接続したい無線 LAN に接続不可能な場所で設定を行う場合は以下の方法で無線 LAN の設定を行います。

Raspberry Pi の USB に無線 LAN ドングルを挿入し設定することで、Raspberry Pi を無線化できます。移動ロボットなどに搭載する際には便利です。

• /etc/network/interfaces の編集

まず、/etc/network/interfaces を以下のように編集します。

```
$ sudo vi /etc/network/interfaces
```

vi : テキストエディタ vi を起動する。

以下を書き換えます。

```
iface wlan0 inet manual
    ↓
iface wlan0 inet dhcp
```

• /etc/wpa_supplicant/wpa_supplicant.conf の編集

次に、無線 LAN の ESSID とキーを登録します。

```
$ sudo bash
# cd /etc/wpa_supplicant
# wpa_passphrase ESSID pass >> wpa_supplicant.conf
```

bash : 標準入力やファイルから読み込んだコマンドを実行する、sh 互換のコマンド言語インタプリタを起動する。

cd : カレントディレクトリを変更する。

SSID には無線 LAN の ESSID、pass には無線 LAN のキーを入力します。リダイレクトの際、> ではなく >> (追記)を使用するように注意してください。結果は以下のようになっていると思います。

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
network={
    ssid="OpenRTM"
    #psk="4332221111"
    psk=142914b76be167767055ff945898baaaf83c42b3ad3b99afb0ae531e8fb15e5e
}
```

無線 LAN アクセスポイントの設定によっては、追加の設定が必要になるかもしれません。以下に、一例を示します。

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
network={
    ssid="OpenRTM"
    proto=WPA2
    key_mgmt=WPA-PSK
    pairwise=TKIP CCMP
    group=TKIP CCMP
    #psk="4332221111"
    psk=142914b76be167767055ff945898baaaf83c42b3ad3b99afb0ae531e8fb15e5e
}
```

最後に、インターフェースを初期化します。

```
# ifdown wlan0
# ifup wlan0
Internet Systems Consortium DHCP Client 4.2.2
Copyright 2004-2011 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
: 中略
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPOFFER from 192.168.11.1
DHCPACK from 192.168.11.1
bound to 192.168.11.26 -- renewal in 34810 seconds.
```

ifdown : ネットワーク・インターフェースを無効にする。

ifup : ネットワーク・インターフェースを有効にする。

ここで、無線 LAN に接続できない場合、`/etc/network/interfaces`, `/etc/wpa_supplicant/wpa_supplicant.conf` の設定を見直してください。

```
# ifconfig wlan0
wlan0    Link encap:Ethernet  HWaddr XX:XX:XX:XX:XX:XX
         inet addr:192.168.11.26  Bcast:192.168.11.255  Mask:255.255.255.0
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:1218 errors:0 dropped:0 overruns:0 frame:0
         TX packets:21 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:250608 (244.7 KiB)  TX bytes:4506 (4.4 KiB)
```

ifconfig : ネットワーク・インタフェースを参照する。

無事、無線 LAN wlan0 に IP アドレスが割り振られました。

3 PC の設定

Bonjour のインストール (Windows のみ)

PC から avahi が設定された RaspberryPi にアクセスするためには、PC 側にも avahi か Bonjour がインストールされている必要があります。

Linux では、RaspberryPi と同様に avahi-daemon をインストールすれば使用できます。また Mac はデフォルトで Bonjour がインストールされているので、特に何もする必要がありません。

Windows ではデフォルトでは Bonjour はインストールされていません。最も簡単に Bonjour を導入する方法は iTunes をインストールすることです。

- [iTunes ダウンロード](#)

どうしても iTunes をインストールしたくない場合は、アーカイバアプリケーションなどで、ダウンロードした iTunesSetup.exe を展開すると BonjourSetup.exe を抽出することもできます。

また、以下の Apple Bonjour 印刷サービスにも Bonjour が同梱されています。(iTunes に同梱されているものよりバージョンが若干古いようです。)

Apple Bonjour [Bonjour 印刷サービス \(v2.0.2.0\)](#)

現在 Apple では Bonjour for Windows 単体としては配布は行なっていませんが、かつて配布していたものを再配布しているサイトも幾つかあります。(ただし、古いバージョンしか入手できないようです。) 以下は Apple サイト以外の Bonjour ダウンロードサイトです。自己責任でご利用ください。

- [BonjourSetup.exe \(v1.0.6.2\)](#)
- [Bonjour64Setup.exe \(v1.0.6.2\)](#)
- [Apple Bonjour SDK \(Apple developer へのログインが必要\)](#)

Bonjour がうまく機能しない場合

ファイアウォールが動作している場合、Bonjour がうまく機能しないことがあります。その場合、UDP ポート 5353 を開放するかファイアウォールを OFF にしてください。

- [ファイアウォールの設定が原因で Bonjour for Windows が動作しない](#)

TeraTerm のインストール

Windows から RaspberryPi に ssh 経由でログインするためには、ssh クライアントをインストールする必要があります。Windows で利用可能なクライアントは多数ありますが、ここでは Tera Term を紹介します。

- [Tera Term](#)



TeraTerm をインストール後、起動すると接続ダイアログが現れるので、先ほど設定したホスト名+.local を「ホスト」のテキストボックスに入力し OK を押します。パスワードが変更されていないならば、

ID: pi

パスワード: raspberry

でログインできます。

なお、Linux や Mac はターミナルウインドウを開いて

```
$ ssh pi@myhost.local
```

ssh : SSH プロトコルを用いてリモート・ホストのコマンドを安全に実行する。

のようにして接続します。

4 Raspberry Pi に OpenRTM-aist をインストール

ここでは RT コンポーネントを開発するために必要なパッケージをインストールしていきます。

Raspberry Pi に開発環境をインストールすれば、Raspberry Pi 上でのセルフコンパイルによる開発が可能です。従って、PC 上の RTCBuilder で生成した RT コンポーネントのコードを、Raspberry Pi 上でコンパイル、実行することができますが、このサイズの組込み Linux ボードでは、通常クロスコンパイルによる開発が普通です。(残念ながら Eclipse をインストールして実用的な速度で使用することは難しいようです)

OpenRTM-aist のインストール

OpenRTM-aist は、apt-get を使用してインストールすることが可能です。

起動した Raspberry Pi にアクセスします。アクセスするのに Tera Term を使用します。

まず以下のコマンドを入力し、sources.list を修正します。

```
$ sudo vi /etc/apt/sources.list
```

vi : テキストエディタ vi を起動する。

sources.list に以下の 1 行を加えてください。

```
deb http://openrtm.org/pub/Linux/raspbian/ jessie main
```

sources.list : パッケージの取得先が記述されているファイル。

次に、以下の手順でインストールを行います。

```
# sudo apt-get update
# sudo apt-get -y --force-yes install gcc g++ make uuid-dev
# sudo apt-get -y --force-yes install libomniorb4-dev omniidl omniorb-nameserver
# sudo apt-get -y --force-yes install openrtm-aist openrtm-aist-dev openrtm-aist-example
# sudo apt-get -y --force-yes install openrtm-aist-python openrtm-aist-python-example
# sudo apt-get -y --force-yes install python-omniorb
```

apt-get : パッケージを取得してインストール/アップデートする。

5 CMake のインストール

CMake は apt-get を利用してインストールする事が可能です。root 権限を持つユーザでインストールを行います。

```
$ sudo apt-get update
$ sudo apt-get -y --force-yes install cmake
```

6 Doxygen のインストール

doxygen を apt-get で以下のようにインストールすることができます。

```
$ sudo apt-get -y --force-yes install doxygen
```

※Doxygen のインストールが完了するまでには、十数分程度かかります。

7 Subversion/Git のインストール

ソースコードを外部から取得する際にたびたび使用するので Subversion/Git をインストールすることをお勧めします。

```
$ sudo apt-get -y --force-yes install subversion git
```

8 OpenCV のインストール

Raspberry Pi で WEB カメラを使用する場合 OpneCV をインストールする必要があります。

```
$sudo apt-get -y --force-yes install libopencv-dev python-opencv
```