

비전기반 물체검출 실습

Turtlebot HOST PC Raspberry Pi를 이용한 Yolo 기반 물체검출 알고리즘 적용 실습

한양대학교

최 준 원

Contents

- Raspberry PI OS Setup
- Raspberry PI initial Setting
- Check if OpenCV is installed & Remove existing Opencv
- Various library installation For OpenCV
- OpenCV install (Build & Compile, Install)
- Darknet install (Modify Makefile, Compile)
- Run YOLO_v3 (Single Image & PI Camera)

Raspberry Pi OS Setup

■ Ubuntu Mate 16.04 설치 시

- Ubuntu Mate 16.04.2 Raspberry Pi용 Image 파일 다운로드

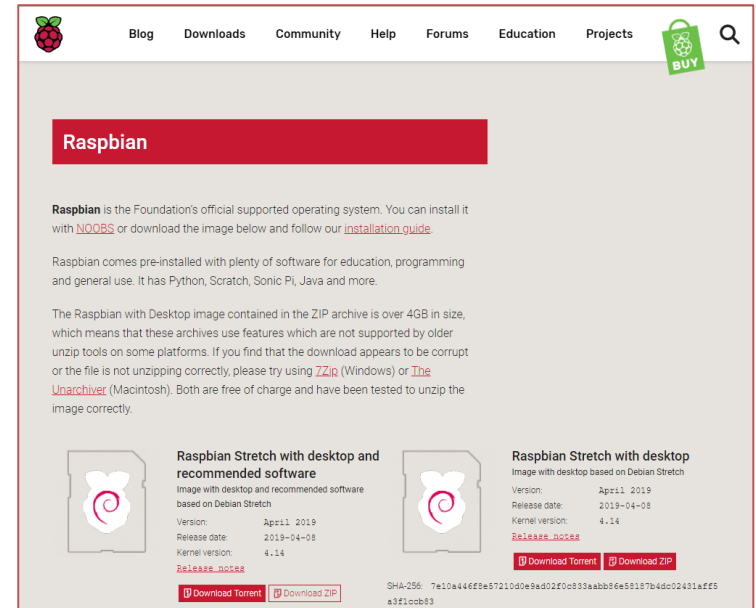
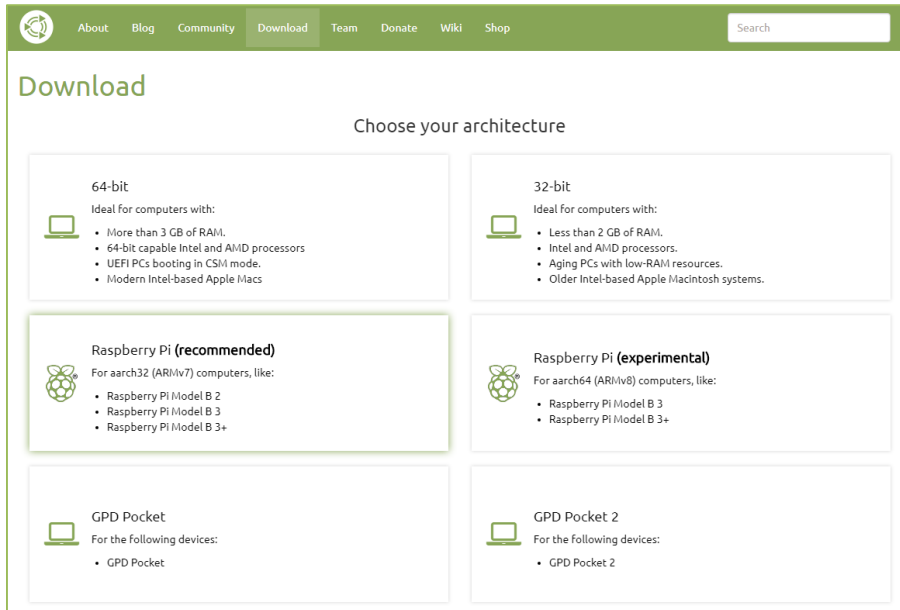
※ 현재 서비스 중인 Ubuntu Mate는 18.04 Beta1으로 미지원 library가 다수 존재, OpenCV 설치 불가.

- Download URL : <https://ubuntu-mate.org/raspberry-pi/ubuntu-mate-16.04.2-desktop-armhf-raspberry-pi.img.xz>

■ Raspbian 설치 시

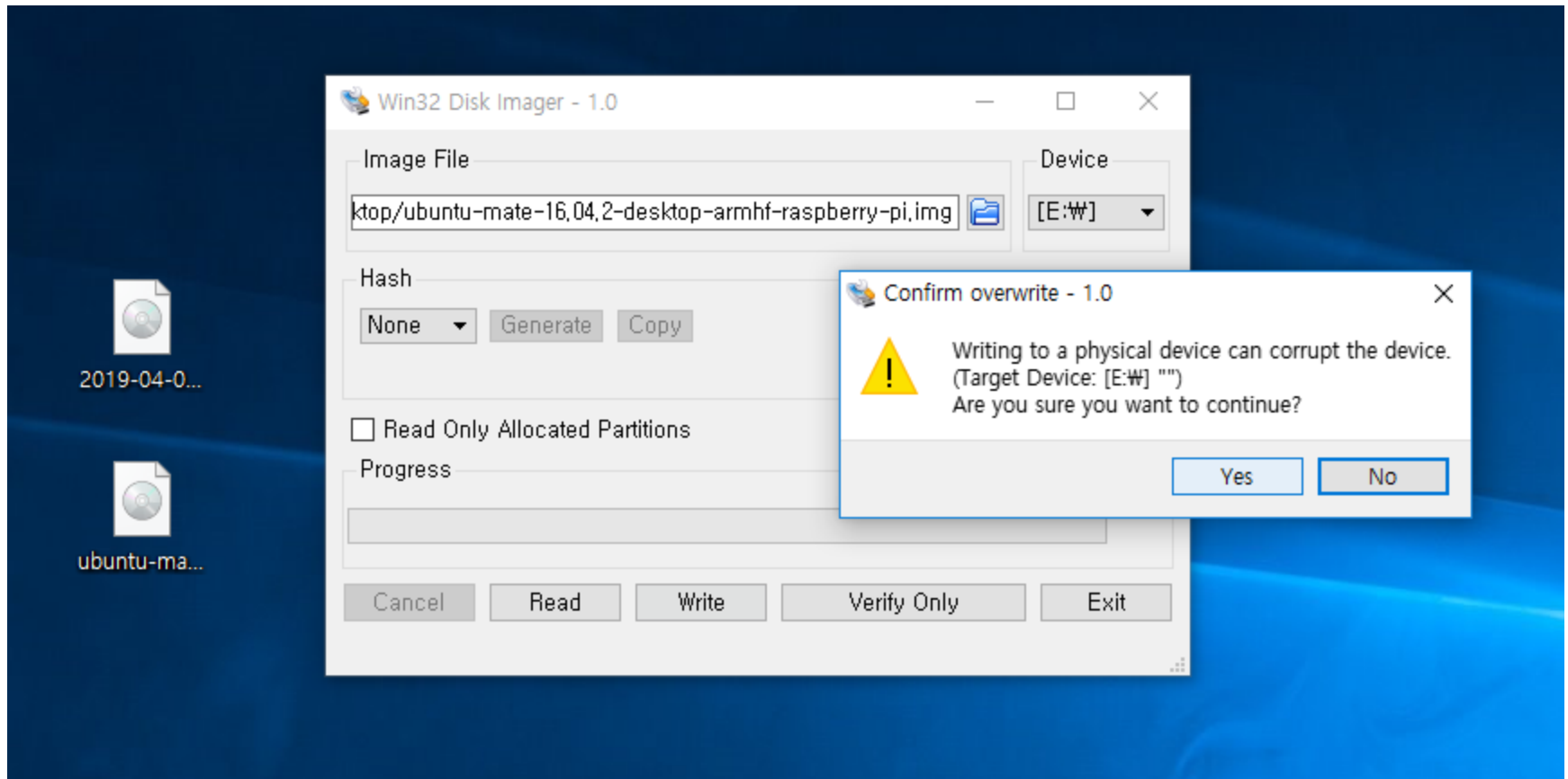
- Raspberry Pi 공식 사이트에서 Image 다운로드

- Download URL : <https://www.raspberrypi.org/downloads/raspbian/>



Raspberry PI OS Setup

- Win32DiskImager 등 프로그램 이용, SD Card로 Image Writing



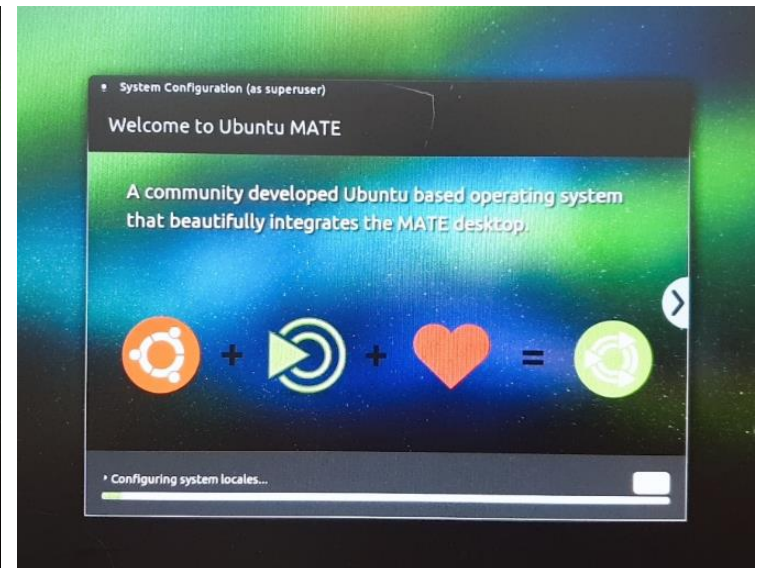
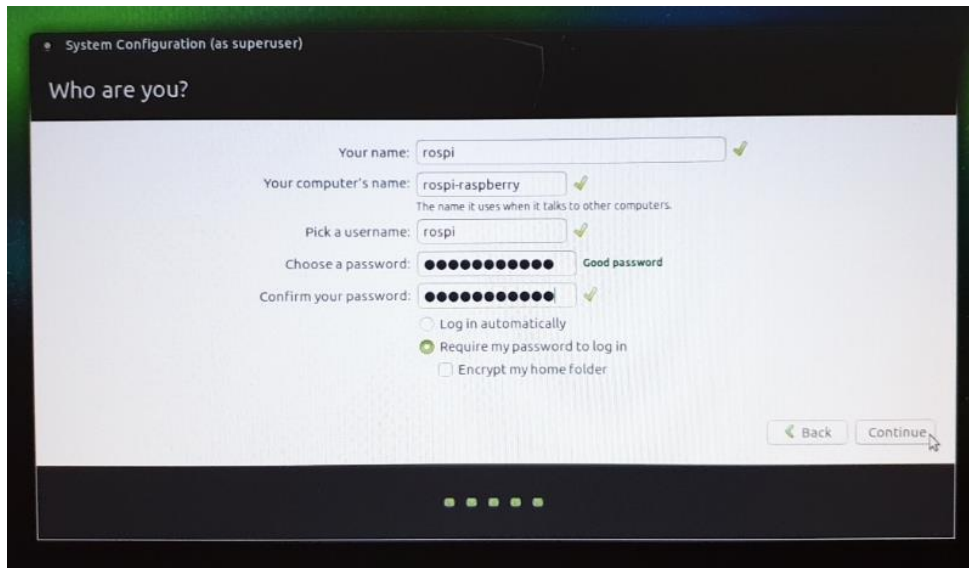
Raspberry PI OS Setup

■ Raspbian 설치 시

- Image writing을 마친 SD카드를 Raspberry PI에 삽입하고 부팅하면 자동설치
- 지역 및 언어는 United State 기준, 계정명 pi로 자동 셋팅 완료
- 따라서 부팅 이후 언어, WIFI, 지역, 키보드 레이아웃 등 별도 설정 필요

■ Ubuntu Mate 16.04 설치 시

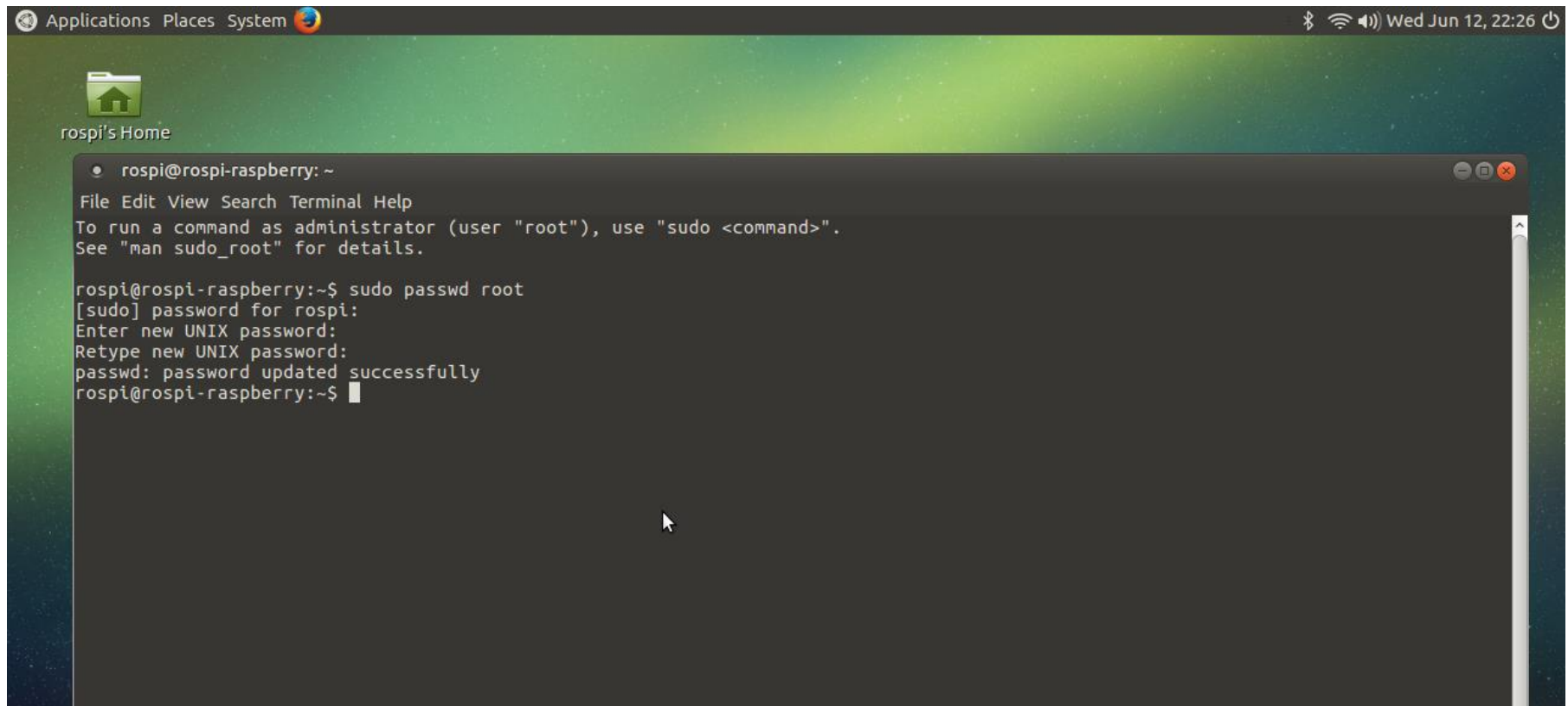
- Image writing을 마친 SD카드를 Raspberry PI에 삽입하고 부팅하면 설치시작
- 언어, WIFI, 지역, 키보드 레이아웃, 계정 등 별도의 정보 입력과정 필요



Raspberry PI initial Setting

■ Root 계정 Password 설정

- 터미널 명령어 : `sudo passwd root`
- Password 입력이 정상적으로 완료되면 Password updated Successfully 확인



The screenshot shows a Raspberry Pi desktop with a green and blue background. A terminal window is open, displaying the following text:

```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
rospi@rospi-raspberry:~$ sudo passwd root  
[sudo] password for rospi:  
Enter new UNIX password:  
Retype new UNIX password:  
passwd: password updated successfully  
rospi@rospi-raspberry:~$
```

Raspberry PI initial Setting

■ Raspberry PI 펌웨어 업데이트

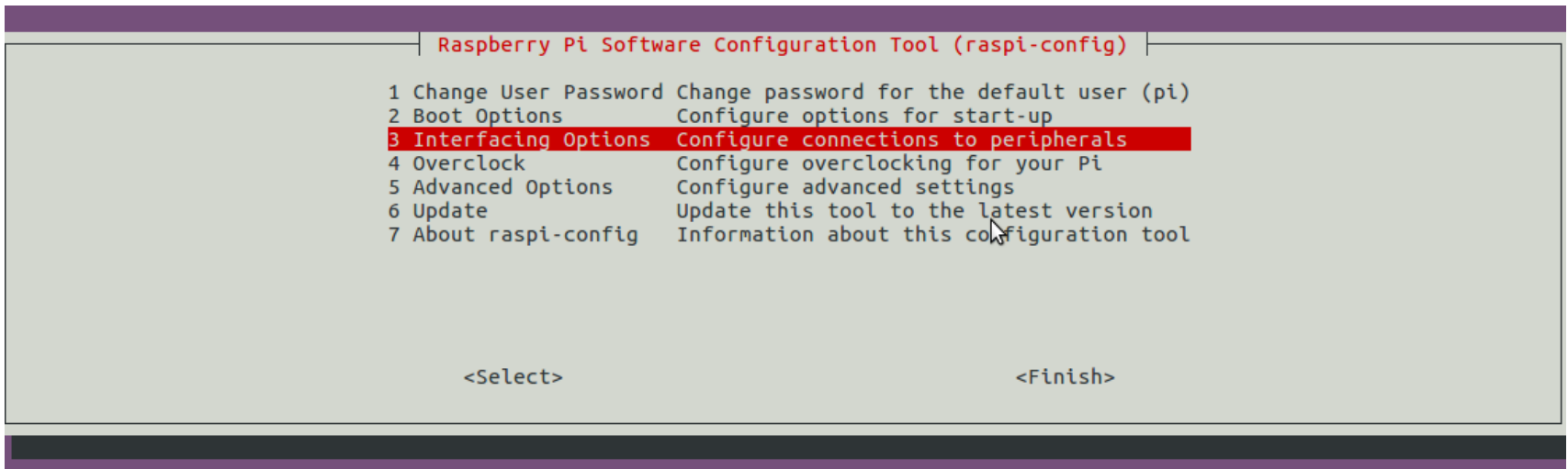
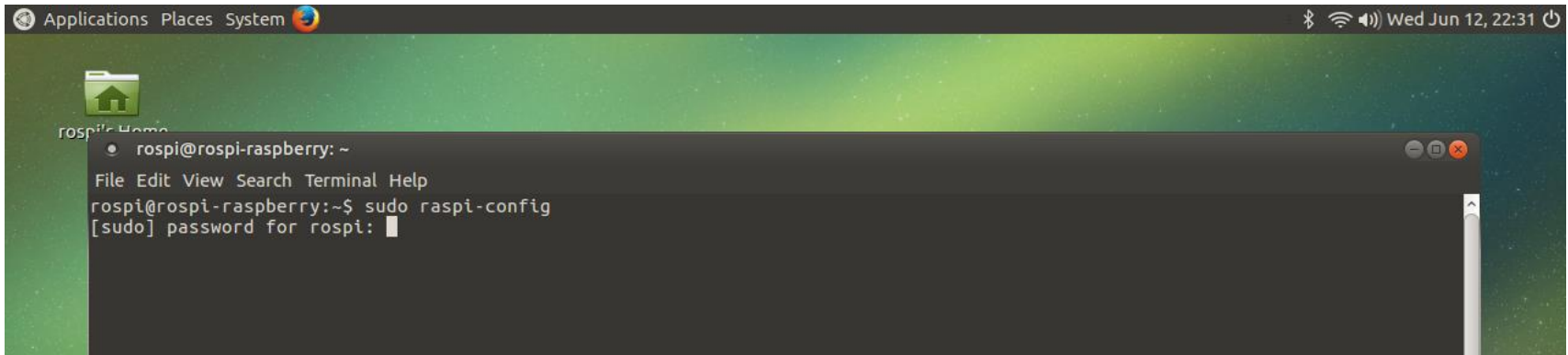
- 명령어 : sudo rpi-update

```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
rospi@rospi-raspberry:~$ sudo rpi-update  
[sudo] password for rospi:  
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom  
*** Performing self-update  
*** Relaunching after update  
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom  
*** We're running for the first time  
*** Backing up files (this will take a few minutes)  
*** Backing up firmware  
*** Backing up modules 4.4.38-v7+  
#####  
WARNING: This update bumps to rpi-4.19.y linux tree  
Be aware there could be compatibility issues with some drivers  
Discussion here:  
https://www.raspberrypi.org/forums/viewtopic.php?f=29&t=224931  
#####  
Would you like to proceed? (y/N)  
*** Downloading specific firmware revision (this will take a few minutes)  
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current  
   Total      100    168    0      0    150      0 --:--:--  0:00:01 --:--:--  150  
100 58.9M  100 58.9M    0     0 1303k      0  0:00:46  0:00:46 --:--:-- 2119k  
*** Updating firmware  
*** Updating kernel modules  
*** depmod 4.19.49-v7+  
*** depmod 4.19.49+  
*** Updating VideoCore libraries  
*** Using HardFP libraries  
*** Updating SDK  
*** Running ldconfig  
*** Storing current firmware revision  
*** Deleting downloaded files  
*** Syncing changes to disk  
*** If no errors appeared, your firmware was successfully updated to ce9a76ebe12cfa9cd76d15be7684af93632365f6  
*** A reboot is needed to activate the new firmware  
rospi@rospi-raspberry:~$
```

Raspberry PI initial Setting

■ Raspberry PI Configuration Tool 설정

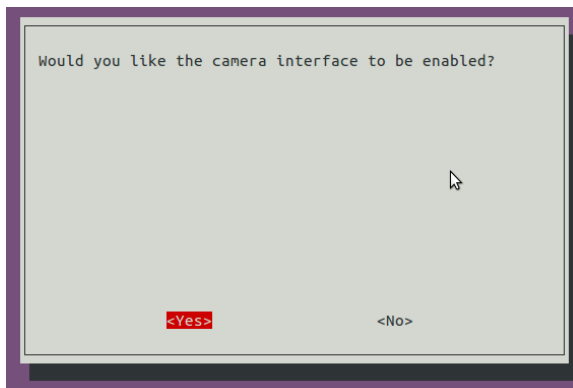
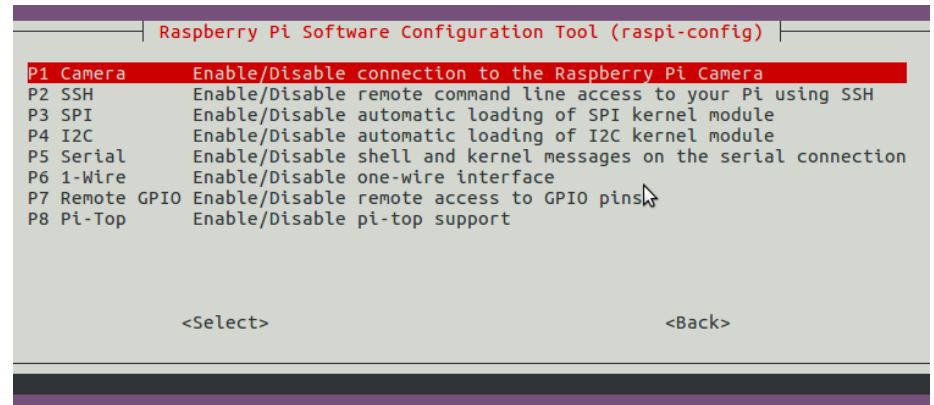
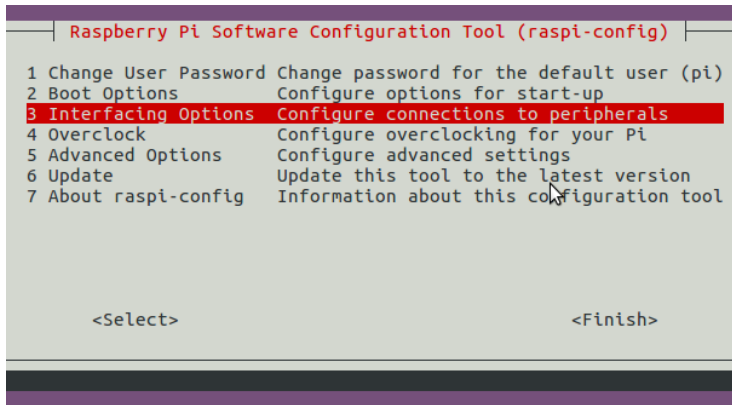
- Configuration Tool 진입 명령어 : `sudo raspi-config`
- Boot 옵션, 입출력 기기사용 활성화, 파일시스템, Update 등 셋팅 가능



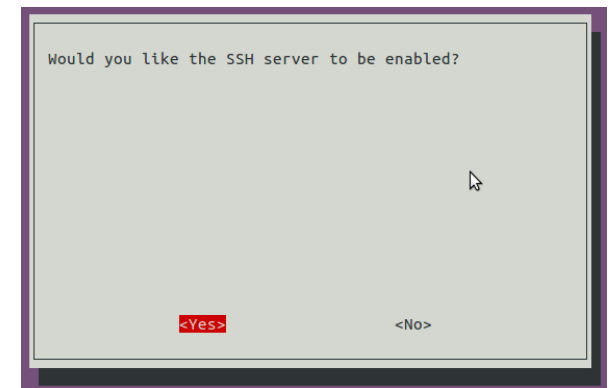
Raspberry PI initial Setting

■ Raspberry PI Configuration Tool 설정(Pi Camera & SSH)

- Interfacing Options 에서 P1 Camera / P2 SSH 를 Enable 상태로 변경



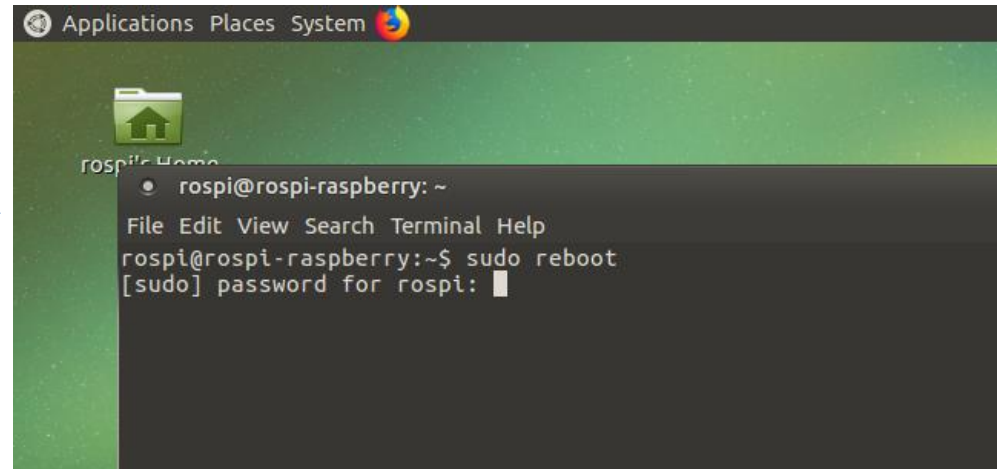
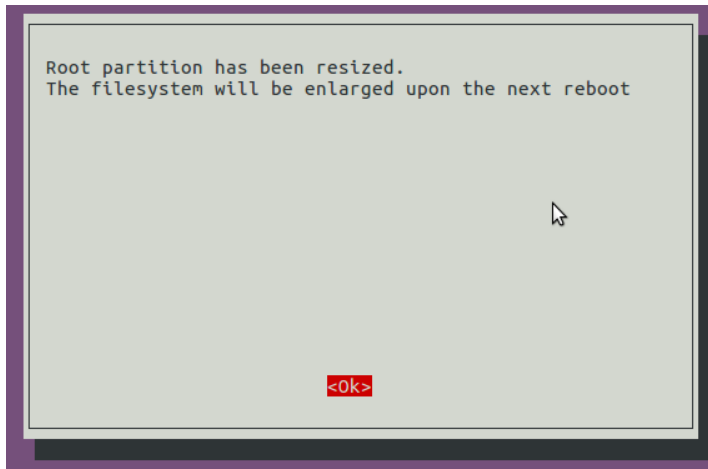
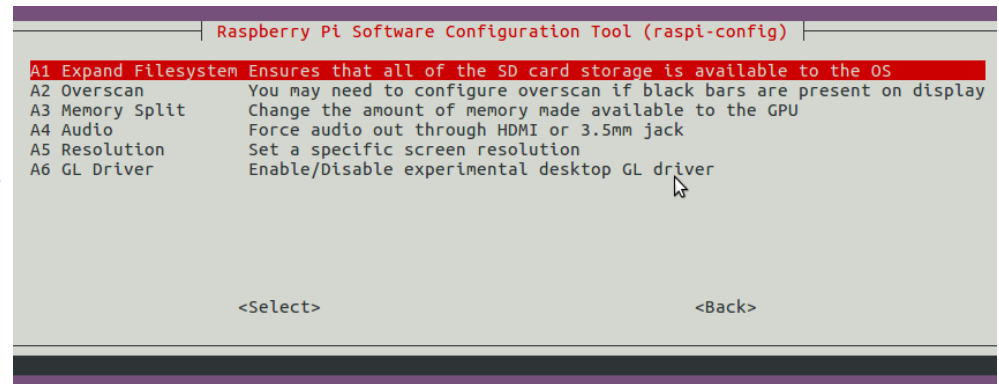
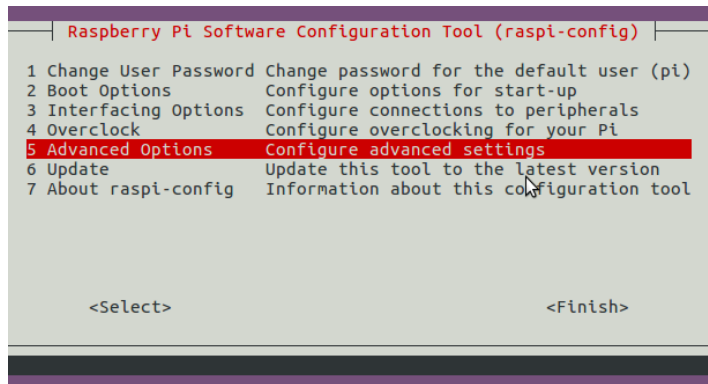
P1 Camera interface를 enable 상태로 변경 후, 위 과정 동일하게 반복. P2 SSH server 기능도 enable 상태로 변경.



Raspberry PI initial Setting

■ Raspberry PI Configuration Tool 설정 (Filesystem 확장)

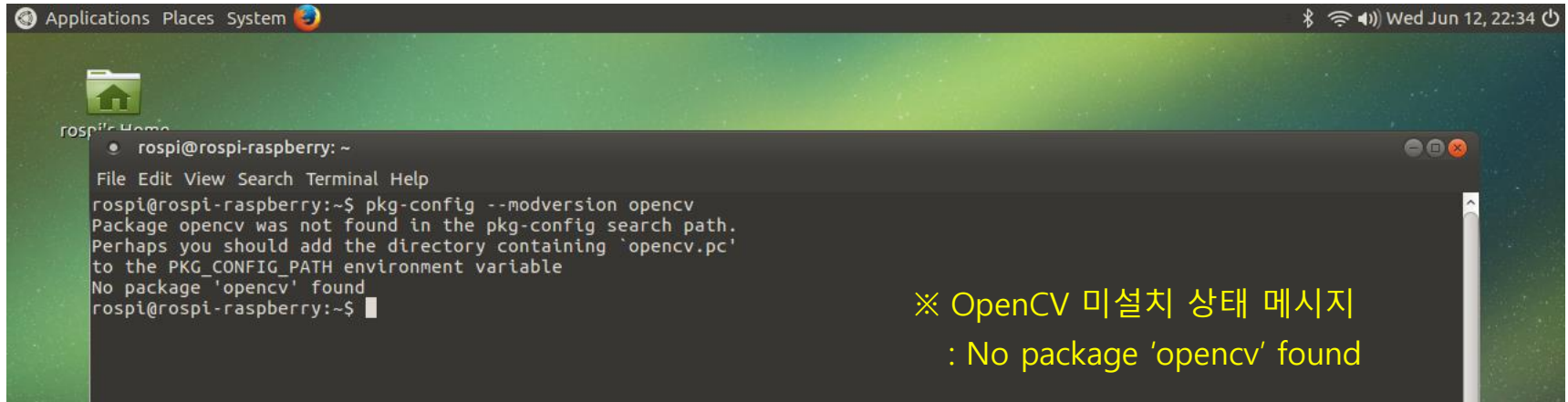
- Advanced Option에서 A1 Expand Filesystem 선택
- Root Partition resized 후 시스템 재시작 (Reboot 명령어 : `sudo reboot`)



Check if OpenCV is installed

■ OpenCV 설치유무 확인

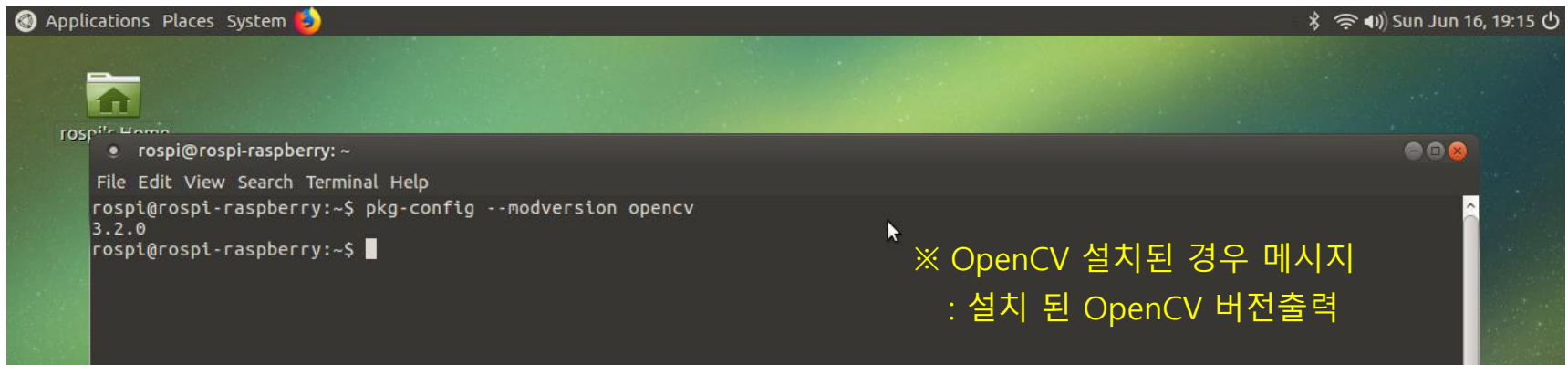
- 명령어 : `pkg-config --modversion opencv`



A terminal window on a Raspberry Pi desktop. The terminal shows the command `pkg-config --modversion opencv` being executed. The output message states that the package 'opencv' was not found in the search path and suggests adding the directory containing 'opencv.pc' to the `PKG_CONFIG_PATH` environment variable. The final line of the output is "No package 'opencv' found".

```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
rospi@rospi-raspberry:~$ pkg-config --modversion opencv  
Package opencv was not found in the pkg-config search path.  
Perhaps you should add the directory containing 'opencv.pc'  
to the PKG_CONFIG_PATH environment variable  
No package 'opencv' found  
rospi@rospi-raspberry:~$
```

※ OpenCV 미설치 상태 메시지
: No package 'opencv' found



A terminal window on a Raspberry Pi desktop. The terminal shows the command `pkg-config --modversion opencv` being executed. The output is the version number "3.2.0".

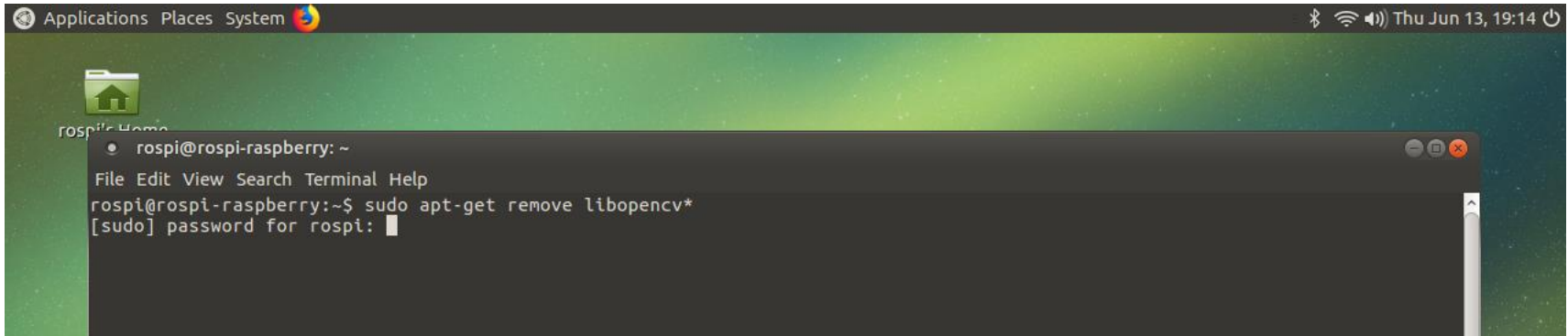
```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
rospi@rospi-raspberry:~$ pkg-config --modversion opencv  
3.2.0  
rospi@rospi-raspberry:~$
```

※ OpenCV 설치된 경우 메시지
: 설치 된 OpenCV 버전출력

Remove existing Opencv

■ 타 버전 OpenCV & 관련 library 삭제

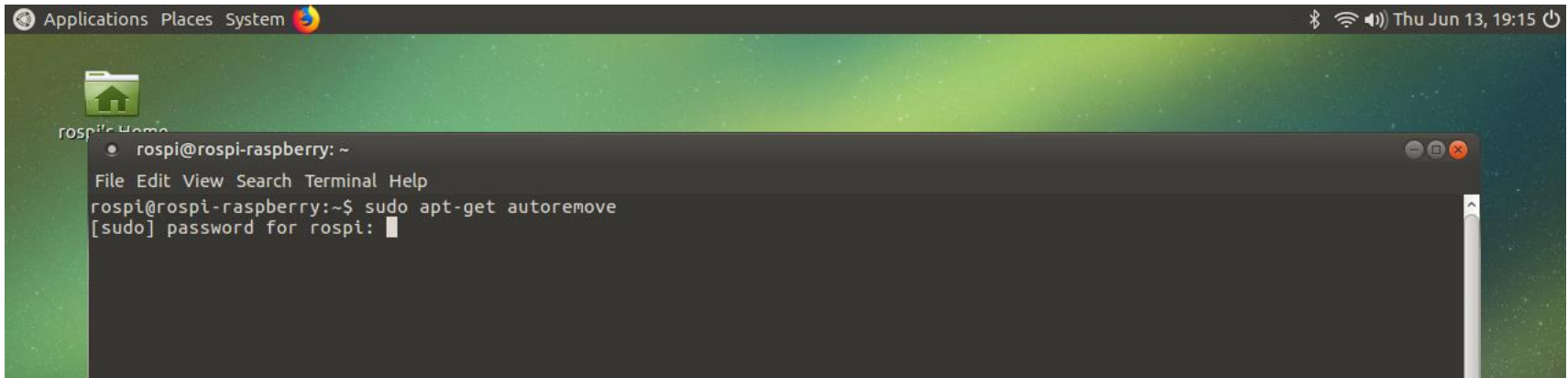
- OpenCV관련 패키지 삭제 - 명령어 : `sudo apt-get remove libopencv*`



A terminal window on a Raspberry Pi desktop. The terminal shows the command `sudo apt-get remove libopencv*` being entered. The prompt is `rospi@rospi-raspberry: ~`. The terminal menu bar includes File, Edit, View, Search, Terminal, and Help. The desktop background is a green and blue gradient. The top bar shows 'Applications Places System' and the date 'Thu Jun 13, 19:14'.

```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
rospi@rospi-raspberry:~$ sudo apt-get remove libopencv*  
[sudo] password for rospi: 
```

- 더 이상 OS에서 사용하지 않는 패키지 삭제 - 명령어 : `sudo apt-get autoremove`



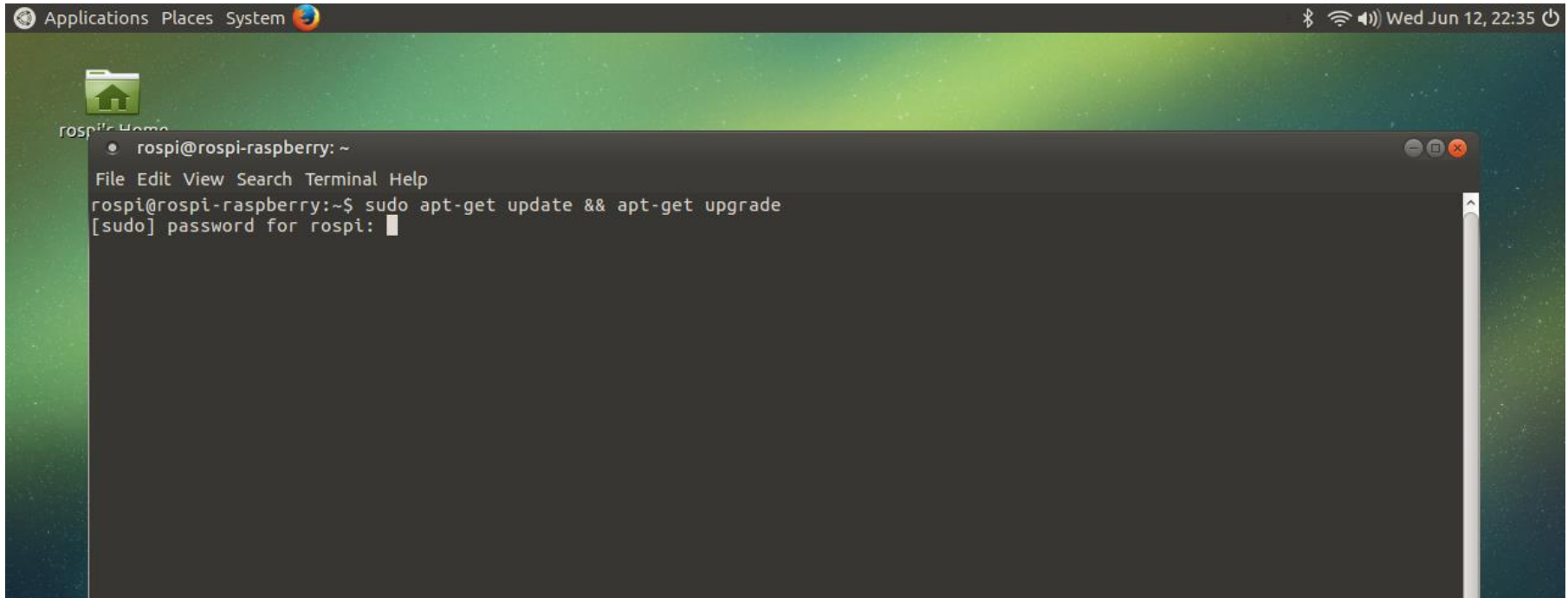
A terminal window on a Raspberry Pi desktop. The terminal shows the command `sudo apt-get autoremove` being entered. The prompt is `rospi@rospi-raspberry: ~`. The terminal menu bar includes File, Edit, View, Search, Terminal, and Help. The desktop background is a green and blue gradient. The top bar shows 'Applications Places System' and the date 'Thu Jun 13, 19:15'.

```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
rospi@rospi-raspberry:~$ sudo apt-get autoremove  
[sudo] password for rospi: 
```

Various Library installation For OpenCV

■ Library 설치 전 System update & upgrade

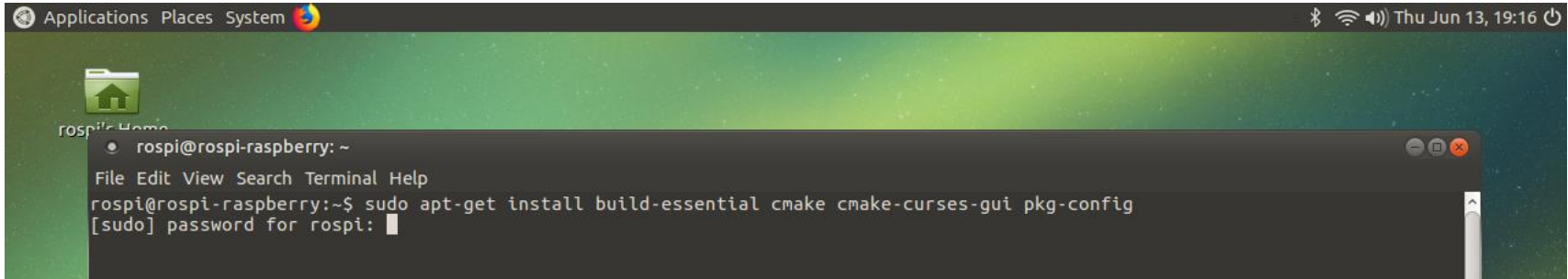
- Apt server Update 및 패키지 Upgrade
 - 명령어 : `sudo apt-get update && apt-get upgrade`



Various Library installation For OpenCV

■ 개발자 도구 설치

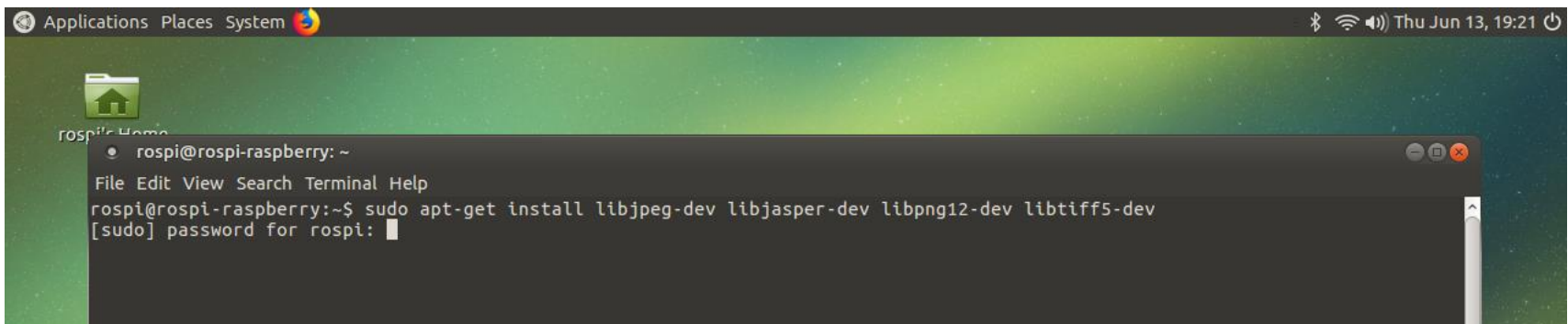
- 명령어 : `sudo apt-get install build-essential cmake cmake-curses-gui pkg-config`



```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
rospi@rospi-raspberry:~$ sudo apt-get install build-essential cmake cmake-curses-gui pkg-config  
[sudo] password for rospi: 
```

■ JPEG, PNG, TIFF 파일 형식 사용을 위한 이미지 처리 I/O 패키지 설치

- 명령어 : `sudo apt-get install libjpeg-dev libjasper-dev libpng12-dev libtiff5-dev`

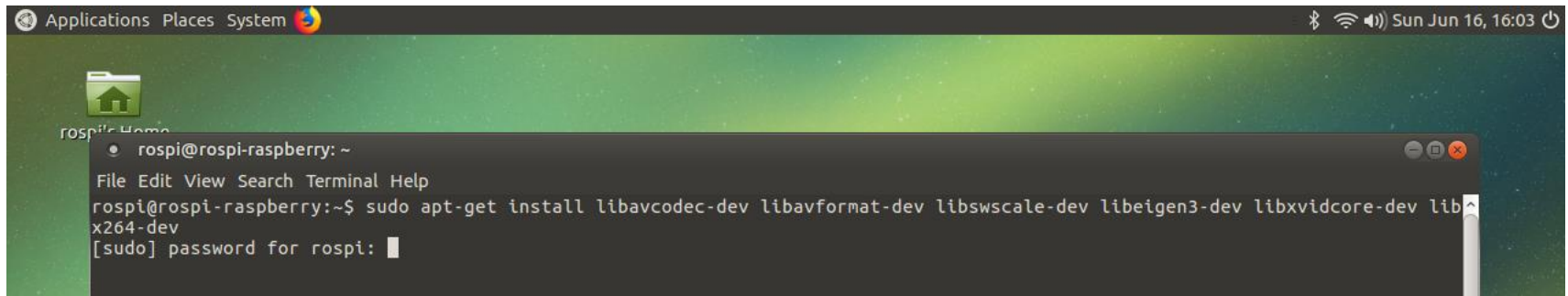


```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
rospi@rospi-raspberry:~$ sudo apt-get install libjpeg-dev libjasper-dev libpng12-dev libtiff5-dev  
[sudo] password for rospi: 
```


Various Library installation For OpenCV

■ 카메라 Stream 작업이 가능하도록 Video I/O 패키지 설치

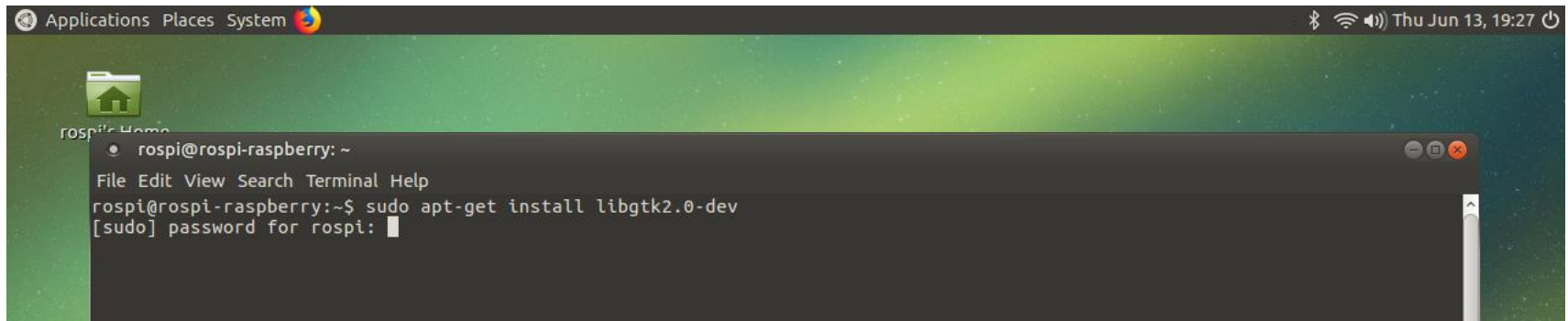
- 명령어 : `sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libeigen3-dev libxvidcore-dev libx264-dev`



A screenshot of a Raspberry Pi desktop environment. A terminal window is open, showing the command `sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libeigen3-dev libxvidcore-dev libx264-dev` being entered. The prompt is `rospi@rospi-raspberry: ~`. The terminal title bar shows `rospi@rospi-raspberry: ~`. The desktop background is a green and blue gradient. The top bar shows `Applications Places System` and the date `Sun Jun 16, 16:03`.

■ OpenCV GUI 프로세싱을 위한 GTK 라이브러리 패키지 설치

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

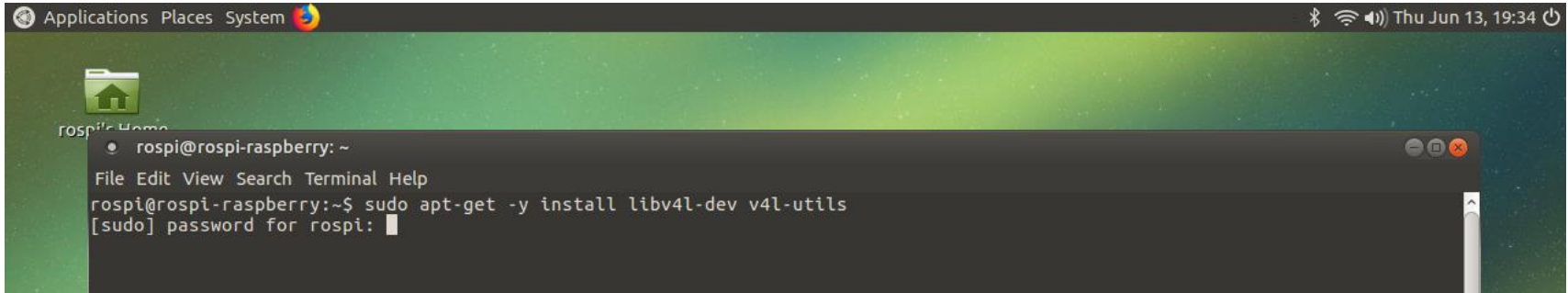


A screenshot of a Raspberry Pi desktop environment. A terminal window is open, showing the command `sudo apt-get install libgtk2.0-dev` being entered. The prompt is `rospi@rospi-raspberry: ~`. The terminal title bar shows `rospi@rospi-raspberry: ~`. The desktop background is a green and blue gradient. The top bar shows `Applications Places System` and the date `Thu Jun 13, 19:27`.

Various Dependency installation For OpenCV

■ 실시간 Video Capture 지원 V4L 라이브러리 및 Utility 패키지 설치

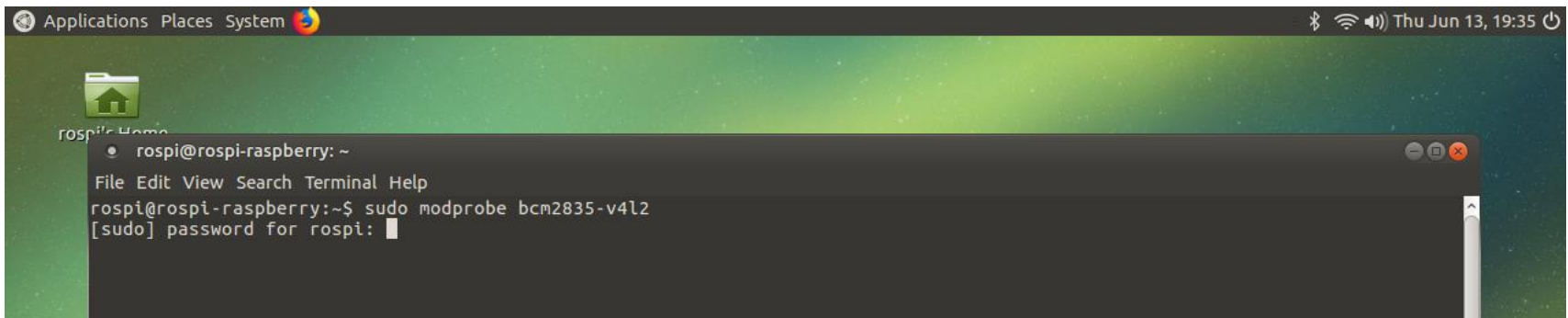
- 명령어 : `sudo apt-get -y install libv4l-dev v4l-utils`



A screenshot of a terminal window on a Raspberry Pi. The window title is "rospi@rospi-raspberry: ~". The terminal shows the command `sudo apt-get -y install libv4l-dev v4l-utils` being entered. The prompt `[sudo]` and the text "password for rospi:" are visible, indicating the command is being executed with root privileges.

■ Linux Kernel module로 V4L 패키지 enable

- 명령어 : `sudo modprobe bcm2835-v4l2`



A screenshot of a terminal window on a Raspberry Pi. The window title is "rospi@rospi-raspberry: ~". The terminal shows the command `sudo modprobe bcm2835-v4l2` being entered. The prompt `[sudo]` and the text "password for rospi:" are visible, indicating the command is being executed with root privileges.

Various Dependency installation For OpenCV

■ OpenCV 기능 최적화 라이브러리 패키지 설치

- 명령어 : `sudo apt-get install libatlas-base-dev gfortran`



A terminal window on a Raspberry Pi desktop. The prompt is `rospi@rospi-raspberry: ~`. The command `sudo apt-get install libatlas-base-dev gfortran` has been entered. The prompt `[sudo] password for rospi:` is shown with a cursor.

■ Python 개발자 도구와 Numpy 라이브러리 설치

- 명령어 : `sudo apt-get install python2.7-dev python-numpy python3-dev python3-numpy`



A terminal window on a Raspberry Pi desktop. The prompt is `rospi@rospi-raspberry: ~`. The command `sudo apt-get install python2.7-dev python-numpy python3-dev python3-numpy` has been entered. The prompt `[sudo] password for rospi:` is shown with a cursor.

OpenCV install

■ OpenCV & OpenCV contrib 다운로드

- Home/user/ 디렉토리 내 opencv 폴더 생성 및 설치파일 다운로드

```
rospi@rospi-rasperry: ~/opencv
File Edit View Search Terminal Help
rospi@rospi-rasperry:~$ mkdir opencv
rospi@rospi-rasperry:~$ cd opencv
rospi@rospi-rasperry:~/opencv$ wget https://github.com/opencv/opencv/archive/3.2.0.zip -O opencv_source.zip
--2019-06-13 19:48:39-- https://github.com/opencv/opencv/archive/3.2.0.zip
Resolving github.com (github.com)... 15.164.81.167
Connecting to github.com (github.com)[15.164.81.167]:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/opencv/opencv/zip/3.2.0 [following]
--2019-06-13 19:48:40-- https://codeload.github.com/opencv/opencv/zip/3.2.0
Resolving codeload.github.com (codeload.github.com)... 192.30.255.120
Connecting to codeload.github.com (codeload.github.com)[192.30.255.120]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: 'opencv_source.zip'

opencv_source.zip          [          ] 78.23M  1.44MB/s  in 51s

2019-06-13 19:49:33 (1.53 MB/s) - 'opencv_source.zip' saved [82033498]

rospi@rospi-rasperry:~/opencv$ wget https://github.com/opencv/opencv_contrib/archive/3.2.0.zip -O opencv_contrib.zip
--2019-06-13 19:50:16-- https://github.com/opencv/opencv_contrib/archive/3.2.0.zip
Resolving github.com (github.com)... 52.78.231.108
Connecting to github.com (github.com)[52.78.231.108]:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/opencv/opencv_contrib/zip/3.2.0 [following]
--2019-06-13 19:50:16-- https://codeload.github.com/opencv/opencv_contrib/zip/3.2.0
Resolving codeload.github.com (codeload.github.com)... 192.30.255.120
Connecting to codeload.github.com (codeload.github.com)[192.30.255.120]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: 'opencv_contrib.zip'

opencv_contrib.zip        [          ] 53.39M  1.98MB/s  in 35s

2019-06-13 19:50:54 (1.52 MB/s) - 'opencv_contrib.zip' saved [55984686]

rospi@rospi-rasperry:~/opencv$
```

- ① opencv 폴더 생성
- ② opencv 폴더 진입
- ③ OpenCV source 파일 다운로드

- 다운로드 완료확인
- ④ OpenCV contrib 파일 다운로드

- 다운로드 완료확인

OpenCV install

■ OpenCV & OpenCV contrib 다운로드 & 압축풀기

- 현재 경로 확인
 - pwd 명령어 사용하여 "/home/계정명" 경로 확인 후, 해당 경로에 폴더 생성 할 것.
- 폴더 생성 명령어
 - ~\$ mkdir opencv
- 폴더 진입 명령어
 - ~\$ cd opencv
- OpenCV Source 파일 다운로드 명령어
 - ~/opencv\$ wget https://github.com/opencv/opencv/archive/3.2.0.zip -O opencv_source.zip
- OpenCV Contrib 파일 다운로드 명령어
 - ~/opencv\$ wget https://github.com/opencv/opencv_contrib/archive/3.2.0.zip -O opencv_contrib.zip
- 다운로드 받은 파일 압축풀기 명령어
 - ~/opencv\$ unzip opencv_source.zip
 - ~/opencv\$ unzip opencv_contrib.zip

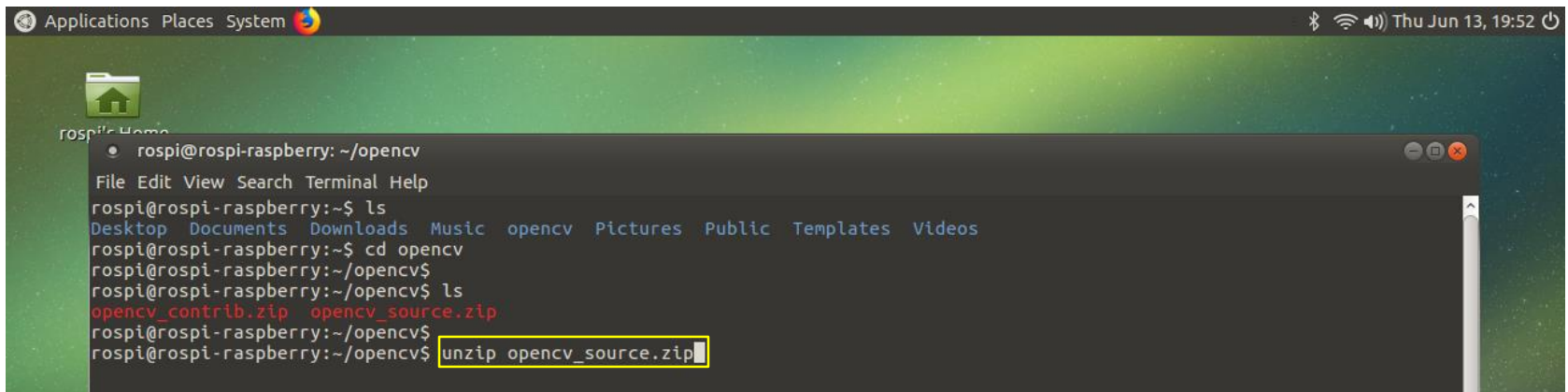
OpenCV install

■ OpenCV & OpenCV contrib 압축풀기

- 다운로드 받은 파일 압축풀기 명령어

~/opencv\$ unzip opencv_source.zip

~/opencv\$ unzip opencv_contrib.zip



```
rospi@rospi-raspberrypi: ~/opencv
File Edit View Search Terminal Help
rospi@rospi-raspberrypi:~$ ls
Desktop Documents Downloads Music opencv Pictures Public Templates Videos
rospi@rospi-raspberrypi:~$ cd opencv
rospi@rospi-raspberrypi:~/opencv$ ls
opencv_contrib.zip opencv_source.zip
rospi@rospi-raspberrypi:~/opencv$ unzip opencv_source.zip
```



```
rospi@rospi-raspberrypi:~/opencv$ ls
opencv-3.2.0 opencv_contrib.zip opencv_source.zip
rospi@rospi-raspberrypi:~/opencv$ unzip opencv_contrib.zip
```

※ 압축 해제 후, ls 명령어로 폴더 생성 확인가능

OpenCV install

■ OpenCV Build 준비

- Build 폴더 생성 및 Build 구성 입력

The terminal window shows the following commands and their corresponding annotations:

- ① 압축해제 확인: `ls` (in `~/opencv`)
- ② source 폴더 진입: `cd opencv-3.2.0`
- ③ 폴더 내용물 확인: `ls` (in `~/opencv/opencv-3.2.0`)
- ④ build 폴더 생성: `mkdir build`
- ⑤ build 폴더 진입: `cd build`
- ⑥ 현재 경로 확인: `pwd`
- ⑦ Build 구성 입력: `cmake -D CMAKE_BUILD_TYPE=RELEASE \`
-D CMAKE_INSTALL_PREFIX=/usr/local \

※ 문장 마지막 \ 입력 후 Enter 입력하면 계속해서 줄 바꿔 이어 입력가능

OpenCV install

■ OpenCV Build 준비

- Build 폴더 생성 명령어 : `~/opencv$ mkdir build`
- 폴더 진입 명령어 : `~/opencv$ cd build`
- 현재 경로확인 명령어 : `~/opencv$ pwd`
- Build 구성 입력 명령어 : `~/opencv$ cmake -D CMAKE_BUILD_TYPE=RELEASE \`
`-D CMAKE_INSTALL_PREFIX=/usr/local \`
`-D BUILD_WITH_DEBUG_INFO=OFF \`
`-D BUILD_DOCS=OFF \`
`-D BUILD_EXAMPLES=OFF \`
`-D BUILD_TESTS=OFF \`
`-D BUILD_opencv_ts=OFF \`
`-D BUILD_PERF_TESTS=OFF \`
`-D INSTALL_C_EXAMPLES=ON \`
`-D INSTALL_PYTHON_EXAMPLES=ON \`
`-D OPENCV_EXTRA_MODULES_PATH=../opencv_contrib-3.2.0/modules \`
`-D ENABLE_NEON=ON \`
`-D WITH_LIBV4L=ON \`
`../`

OpenCV install

■ OpenCV Build 완료

```
rospi@rospi-raspberry: ~/opencv/opencv-3.2.0/build
File Edit View Search Terminal Help
-- Include path: /home/rospi/opencv/opencv-3.2.0/3rdparty/include/opencv/1.2
-- Use AMDFFT: NO
-- Use AMDBLAS: NO
--
-- Python 2:
-- Interpreter: /usr/bin/python2.7 (ver 2.7.12)
-- Libraries: /usr/lib/arm-linux-gnueabi/libpython2.7.so (ver 2.7.12)
-- numpy: /usr/lib/python2.7/dist-packages/numpy/core/include (ver 1.11.0)
-- packages path: lib/python2.7/dist-packages
--
-- Python 3:
-- Interpreter: /usr/bin/python3 (ver 3.5.2)
-- Libraries: /usr/lib/arm-linux-gnueabi/libpython3.5m.so (ver 3.5.2)
-- numpy: /usr/lib/python3/dist-packages/numpy/core/include (ver 1.11.0)
-- packages path: lib/python3.5/dist-packages
--
-- Python (for build): /usr/bin/python2.7
--
-- Java:
-- ant: NO
-- JNI: NO
-- Java wrappers: NO
-- Java tests: NO
--
-- Matlab: Matlab not found or implicitly disabled
--
-- Tests and samples:
-- Tests: NO
-- Performance tests: NO
-- C/C++ Examples: NO
--
-- Install path: /usr/local
--
-- cvconfig.h is in: /home/rospi/opencv/opencv-3.2.0/build
--
--
-- Configuring done
-- Generating done
-- Build files have been written to: /home/rospi/opencv/opencv-3.2.0/build
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$
```

※ 정상적인 Build 완료 시
왼쪽 박스 메시지 확인가능

OpenCV install

■ OpenCV Compile 시작

- Compile 명령어 : make
- /home/계정명/opencv/opencv-3.2.0/build 디렉토리 내에서 실행

```
rospi@rospi-raspberry: ~/opencv/opencv-3.2.0/build
File Edit View Search Terminal Help
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$ ls
3rdparty      CMakeFiles      CPackSourceConfig.cmake  doc      modules      samples
apps          cmake_install.cmake  CTestTestfile.cmake     include  opencv2      text_config.hpp
bin           cmake_uninstall.cmake custom_hal.hpp           junk     OpenCVConfig.cmake  unix-install
carotene      CMakeVars.txt      cvconfig.h              lib      OpenCVConfig-version.cmake  version_string.tmp
CMakeCache.txt CPackConfig.cmake  data                    Makefile  OpenCVModules.cmake

rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$ make
Scanning dependencies of target libwebp
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/layer.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/quant.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/vp8.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/webp.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/idec.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/tree.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/buffer.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/vp8l.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/frame.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/io.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/alpha.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dsp/yuv.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dsp/upsampling.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dsp/upsampling_neon.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dsp/cpu.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dsp/enc.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dsp/lossless.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dsp/dec_sse2.c.o
```

① Build 완료 확인

② Compile 시작

※ Compile 완료까지
약 3시간 소요

※ 1GB로 Swap Memory 공간 확장 후, make -j4 명령어 사용 시에는 Compile 시간 단축 가능
그러나, Ubuntu mate에서는 Allocate memory Error 발생으로 make 명령어 사용을 추천함.

OpenCV install

■ OpenCV Compile 완료

```
rospi@rospi-raspberry: ~/opencv/opencv-3.2.0/build
File Edit View Search Terminal Help
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/haarfeatures.cpp.o
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/cascadeclassifier.cpp.o
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/features.cpp.o
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/imagestorage.cpp.o
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/old_ml_tree.cpp.o
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/boost.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/traincascade.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/old_ml_data.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/HOGfeatures.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/lbpfeatures.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/old_ml_boost.cpp.o
[100%] Linking CXX executable ../../bin/opencv_traincascade
[100%] Built target opencv_traincascade
Scanning dependencies of target opencv_createsamples
[100%] Building CXX object apps/createsamples/CMakeFiles/opencv_createsamples.dir/utility.cpp.o
[100%] Building CXX object apps/createsamples/CMakeFiles/opencv_createsamples.dir/createsamples.cpp.o
[100%] Linking CXX executable ../../bin/opencv_createsamples
[100%] Built target opencv_createsamples
Scanning dependencies of target opencv_annotation
[100%] Building CXX object apps/annotation/CMakeFiles/opencv_annotation.dir/opencv_annotation.cpp.o
[100%] Linking CXX executable ../../bin/opencv_annotation
[100%] Built target opencv_annotation
Scanning dependencies of target opencv_visualisation
[100%] Building CXX object apps/visualisation/CMakeFiles/opencv_visualisation.dir/opencv_visualisation.cpp.o
[100%] Linking CXX executable ../../bin/opencv_visualisation
[100%] Built target opencv_visualisation
Scanning dependencies of target opencv_interactive-calibration
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/calibController.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/parametersController.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/rotationConverters.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/frameProcessor.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/calibPipeline.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/main.cpp.o
[100%] Linking CXX executable ../../bin/opencv_interactive-calibration
[100%] Built target opencv_interactive-calibration
Scanning dependencies of target opencv_version
[100%] Building CXX object apps/version/CMakeFiles/opencv_version.dir/opencv_version.cpp.o
[100%] Linking CXX executable ../../bin/opencv_version
[100%] Built target opencv_version
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$
```

※ 정상적인 Compile 완료 시
왼쪽 박스 메시지 확인가능

OpenCV install

■ OpenCV Compile 완료 확인 및 Make install

- 명령어 : `sudo make install`

```
rospi@rospi-raspberrypi: ~/opencv/opencv-3.2.0/build
File Edit View Search Terminal Help
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/imagestorage.cpp.o
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/old_ml_tree.cpp.o
[ 99%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/boost.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/traincascade.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/old_ml_data.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/HOGfeatures.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/lbpfeatures.cpp.o
[100%] Building CXX object apps/traincascade/CMakeFiles/opencv_traincascade.dir/old_ml_boost.cpp.o
[100%] Linking CXX executable ../../bin/opencv_traincascade
[100%] Built target opencv_traincascade
Scanning dependencies of target opencv_createsamples
[100%] Building CXX object apps/createsamples/CMakeFiles/opencv_createsamples.dir/utility.cpp.o
[100%] Building CXX object apps/createsamples/CMakeFiles/opencv_createsamples.dir/createsamples.cpp.o
[100%] Linking CXX executable ../../bin/opencv_createsamples
[100%] Built target opencv_createsamples
Scanning dependencies of target opencv_annotation
[100%] Building CXX object apps/annotation/CMakeFiles/opencv_annotation.dir/opencv_annotation.cpp.o
[100%] Linking CXX executable ../../bin/opencv_annotation
[100%] Built target opencv_annotation
Scanning dependencies of target opencv_visualisation
[100%] Building CXX object apps/visualisation/CMakeFiles/opencv_visualisation.dir/opencv_visualisation.cpp.o
[100%] Linking CXX executable ../../bin/opencv_visualisation
[100%] Built target opencv_visualisation
Scanning dependencies of target opencv_interactive-calibration
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/calibController.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/parametersController.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/rotationConverters.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/frameProcessor.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/calibPipeline.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/main.cpp.o
[100%] Linking CXX executable ../../bin/opencv_interactive-calibration
[100%] Built target opencv_interactive-calibration
Scanning dependencies of target opencv_version
[100%] Building CXX object apps/version/CMakeFiles/opencv_version.dir/opencv_version.cpp.o
[100%] Linking CXX executable ../../bin/opencv_version
[100%] Built target opencv_version
rospi@rospi-raspberrypi:~/opencv/opencv-3.2.0/build$
rospi@rospi-raspberrypi:~/opencv/opencv-3.2.0/build$
rospi@rospi-raspberrypi:~/opencv/opencv-3.2.0/build$ sudo make install
[sudo] password for rospi: 
```

※ 정상적인 Compile 완료 시
왼쪽 박스 메시지 확인가능

OpenCV install

OpenCV install

- OpenCV install 완료 확인 및 cache 갱신
 - 명령어 : `sudo ldconfig`

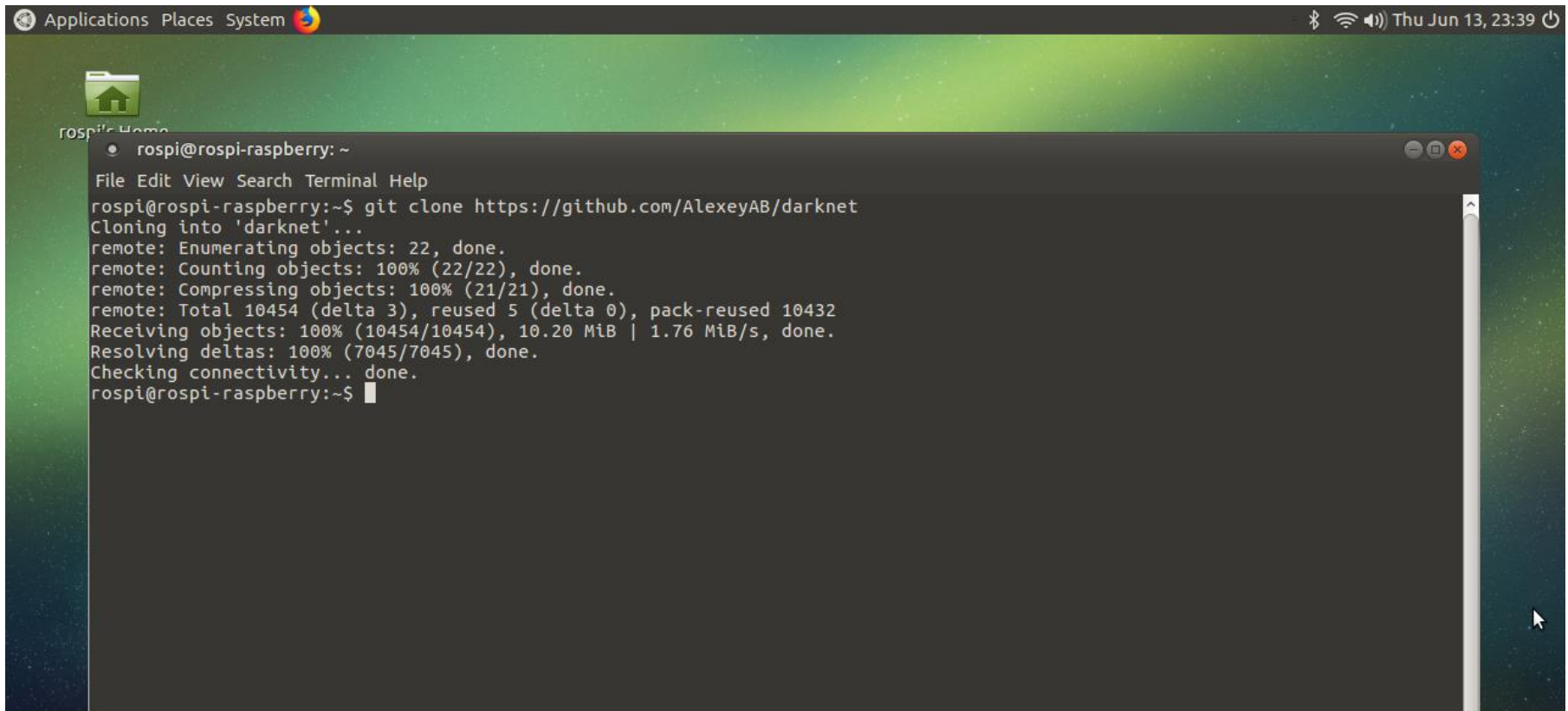
```
rospi@rospi-raspberry: ~/opencv/opencv-3.2.0/build
File Edit View Search Terminal Help
-- Installing: /usr/local/share/OpenCV/samples/python/houghlines.py
-- Installing: /usr/local/share/OpenCV/samples/python/lk_homography.py
-- Installing: /usr/local/share/OpenCV/samples/python/stereo_match.py
-- Installing: /usr/local/share/OpenCV/samples/python/letter_recog.py
-- Installing: /usr/local/share/OpenCV/samples/python/watershed.py
-- Installing: /usr/local/share/OpenCV/samples/python/floodfill.py
-- Installing: /usr/local/share/OpenCV/samples/python/_coverage.py
-- Installing: /usr/local/share/OpenCV/samples/python/digits_video.py
-- Installing: /usr/local/share/OpenCV/samples/python/browse.py
-- Installing: /usr/local/share/OpenCV/samples/python/lappyr.py
-- Installing: /usr/local/share/OpenCV/samples/python/distrans.py
-- Installing: /usr/local/share/OpenCV/samples/python/lk_track.py
-- Installing: /usr/local/share/OpenCV/samples/python/digits_adjust.py
-- Installing: /usr/local/share/OpenCV/samples/python/calibrate.py
-- Installing: /usr/local/share/OpenCV/samples/python/opencv_version.py
-- Installing: /usr/local/share/OpenCV/samples/python/morphology.py
-- Installing: /usr/local/share/OpenCV/samples/python/facedetect.py
-- Installing: /usr/local/share/OpenCV/samples/python/video.py
-- Installing: /usr/local/share/OpenCV/samples/python/kmeans.py
-- Installing: /usr/local/share/OpenCV/samples/python/common.py
-- Installing: /usr/local/share/OpenCV/samples/python/grabcut.py
-- Installing: /usr/local/share/OpenCV/samples/python/coherence.py
-- Installing: /usr/local/share/OpenCV/samples/python/kalman.py
-- Installing: /usr/local/share/OpenCV/samples/python/houghcircles.py
-- Installing: /usr/local/share/OpenCV/samples/python/asift.py
-- Installing: /usr/local/share/OpenCV/samples/python/tst_scene_render.py
-- Installing: /usr/local/share/OpenCV/samples/python/digits.py
-- Installing: /usr/local/share/OpenCV/samples/python/plane_ar.py
-- Installing: /usr/local/share/OpenCV/samples/python/turing.py
-- Installing: /usr/local/share/OpenCV/samples/python/fitline.py
-- Installing: /usr/local/share/OpenCV/samples/python/gabor_threads.py
-- Installing: /usr/local/share/OpenCV/samples/python/plane_tracker.py
-- Installing: /usr/local/share/OpenCV/samples/python/dft.py
-- Installing: /usr/local/share/OpenCV/samples/python/video_threaded.py
-- Installing: /usr/local/share/OpenCV/samples/python/edge.py
-- Installing: /usr/local/share/OpenCV/samples/python/mouse_and_match.py
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$
rospi@rospi-raspberry:~/opencv/opencv-3.2.0/build$ sudo ldconfig
```

Cache 갱신

Darknet install

■ Darknet download

- 명령어 : git clone <https://github.com/AlexeyAB/darknet>
- Git 패키지가 설치되어 있지 않다면, sudo apt install git 명령어 사용하여 설치

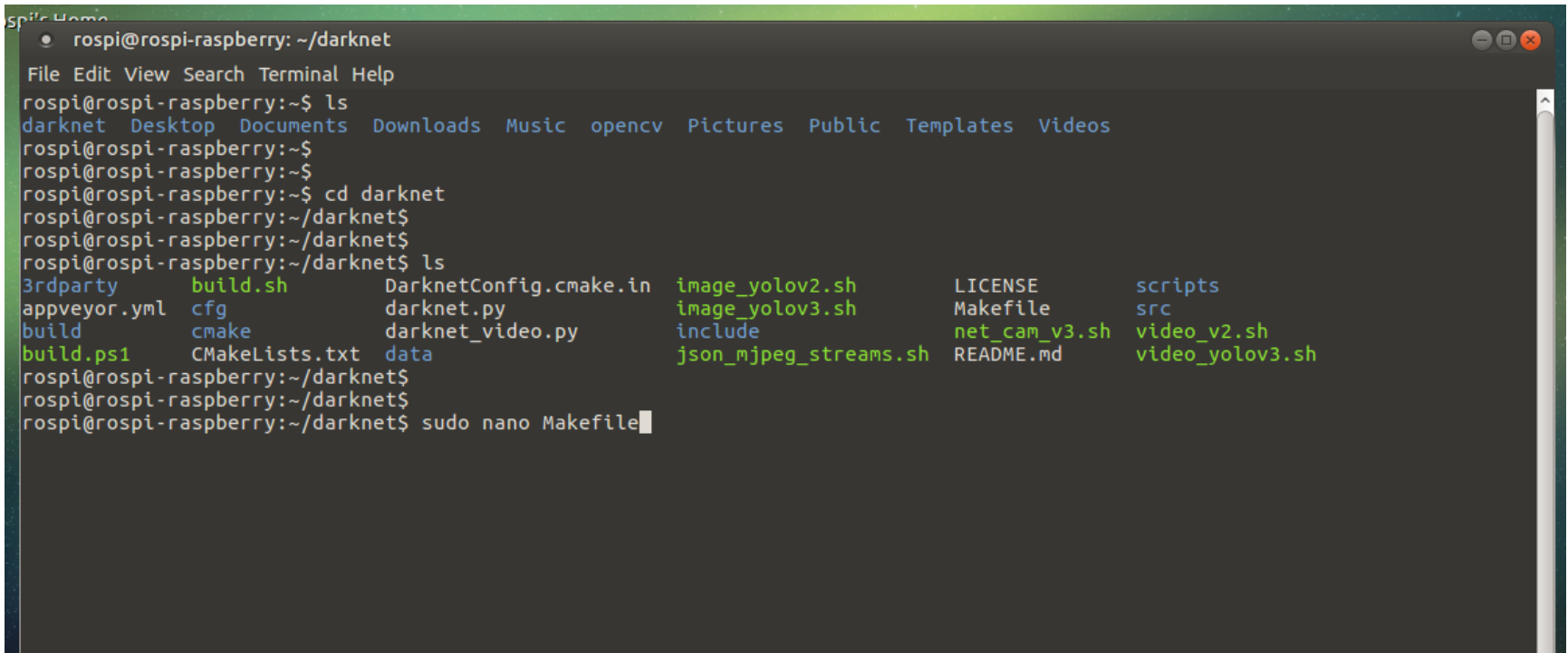


```
rospi@rospi-raspberry: ~  
File Edit View Search Terminal Help  
rospi@rospi-raspberry:~$ git clone https://github.com/AlexeyAB/darknet  
Cloning into 'darknet'...  
remote: Enumerating objects: 22, done.  
remote: Counting objects: 100% (22/22), done.  
remote: Compressing objects: 100% (21/21), done.  
remote: Total 10454 (delta 3), reused 5 (delta 0), pack-reused 10432  
Receiving objects: 100% (10454/10454), 10.20 MiB | 1.76 MiB/s, done.  
Resolving deltas: 100% (7045/7045), done.  
Checking connectivity... done.  
rospi@rospi-raspberry:~$
```

Darknet install

■ Makefile 수정

- Darknet 폴더 진입 후, ls 명령어로 파일 다운로드 정상여부 확인
- Nano 편집기 사용 Makefile 열기 명령어 : `sudo nano Makefile`
- Nano 패키지가 설치되어 있지 않다면, `sudo apt install nano` 명령어 사용 설치

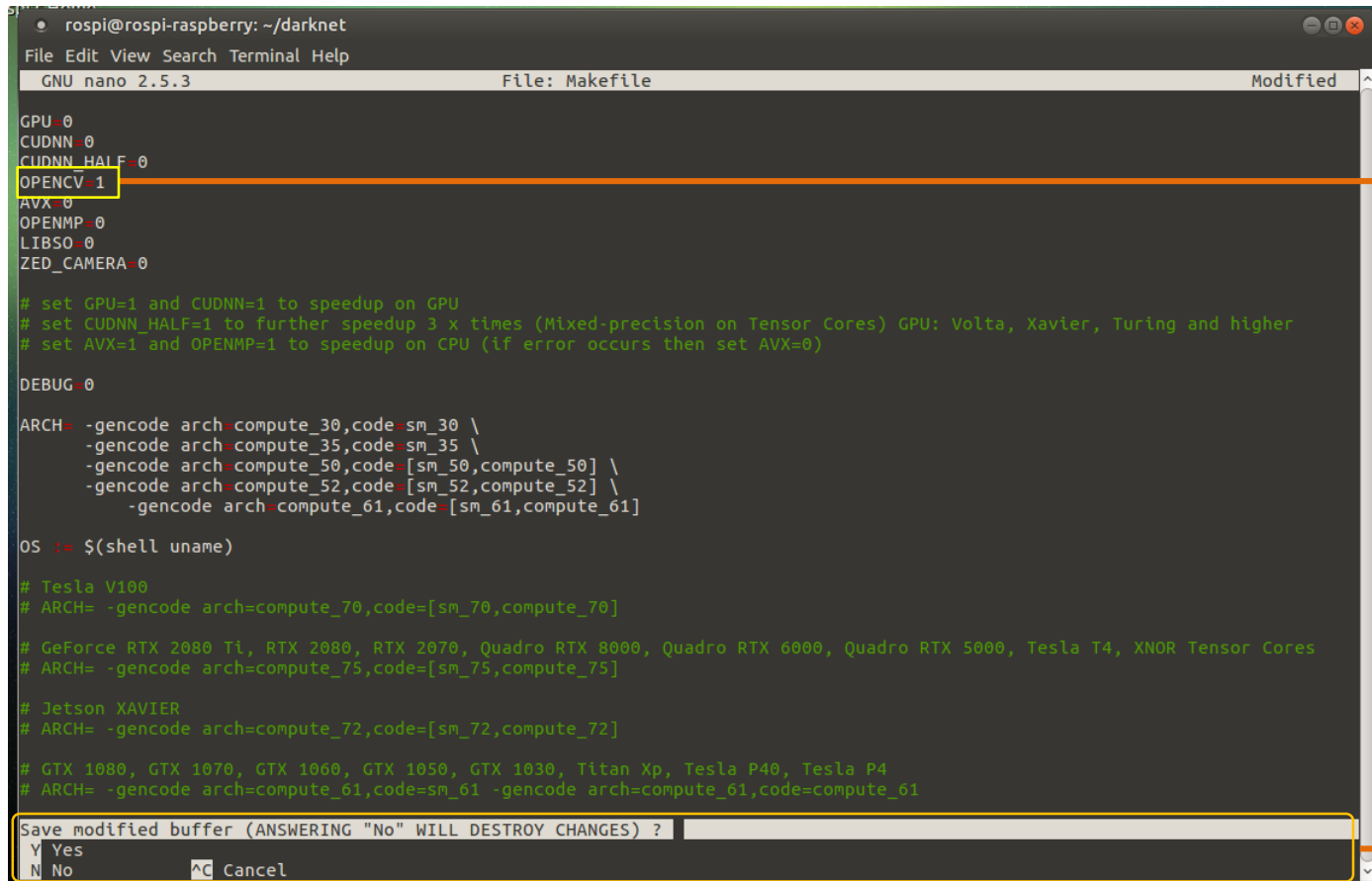


```
rospi@rospi-raspberry: ~/darknet
File Edit View Search Terminal Help
rospi@rospi-raspberry:~$ ls
darknet Desktop Documents Downloads Music opencv Pictures Public Templates Videos
rospi@rospi-raspberry:~$
rospi@rospi-raspberry:~$
rospi@rospi-raspberry:~$ cd darknet
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ ls
3rdparty      build.sh      DarknetConfig.cmake.in  image_yolov2.sh  LICENSE      scripts
appveyor.yml  cfg           darknet.py              image_yolov3.sh  Makefile     src
build         cmake         darknet_video.py        include           net_cam_v3.sh video_v2.sh
build.ps1    CMakeLists.txt data                json_mjpeg_streams.sh  README.md    video_yolov3.sh
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ sudo nano Makefile
```

Darknet install

■ Makefile 수정

- Makefile 내 5-Line의 OPENCV=0을 OPENCV=1로 화면과 같이 수정
- 수정 후, Ctrl + x 단축키 사용 y를 눌러 저장하고 터미널로 돌아간다.



```
rospi@rospi-raspberry: ~/darknet
File Edit View Search Terminal Help
GNU nano 2.5.3 File: Makefile Modified
GPU=0
CUDNN=0
CUDNN_HALF=0
OPENCV=0
AVX=0
OPENMP=0
LIBS0=0
ZED_CAMERA=0

# set GPU=1 and CUDNN=1 to speedup on GPU
# set CUDNN_HALF=1 to further speedup 3 x times (Mixed-precision on Tensor Cores) GPU: Volta, Xavier, Turing and higher
# set AVX=1 and OPENMP=1 to speedup on CPU (if error occurs then set AVX=0)

DEBUG=0

ARCH= -gencode arch=compute_30,code=sm_30 \
      -gencode arch=compute_35,code=sm_35 \
      -gencode arch=compute_50,code=[sm_50,compute_50] \
      -gencode arch=compute_52,code=[sm_52,compute_52] \
      -gencode arch=compute_61,code=[sm_61,compute_61]

OS := $(shell uname)

# Tesla V100
# ARCH= -gencode arch=compute_70,code=[sm_70,compute_70]

# GeForce RTX 2080 Ti, RTX 2080, RTX 2070, Quadro RTX 8000, Quadro RTX 6000, Quadro RTX 5000, Tesla T4, XNOR Tensor Cores
# ARCH= -gencode arch=compute_75,code=[sm_75,compute_75]

# Jetson XAVIER
# ARCH= -gencode arch=compute_72,code=[sm_72,compute_72]

# GTX 1080, GTX 1070, GTX 1060, GTX 1050, GTX 1030, Titan Xp, Tesla P40, Tesla P4
# ARCH= -gencode arch=compute_61,code=sm_61 -gencode arch=compute_61,code=compute_61

Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
Y Yes
N No ^C Cancel
```

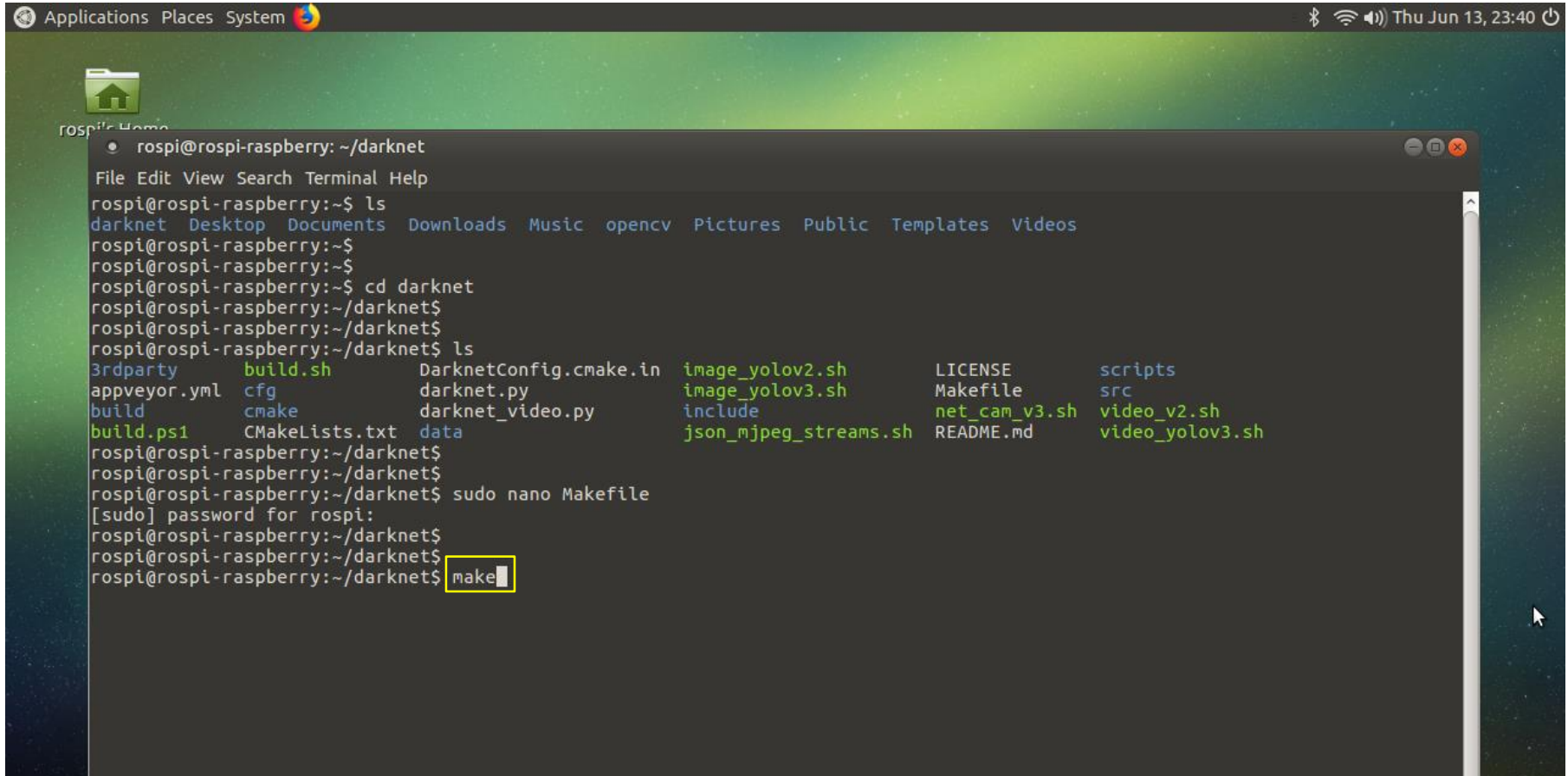
① OPENCV = 0 을
OPENCV = 1 로
수정

② 파일 저장 후 종료

Darknet install

■ Darknet Make install

- Darknet 디렉토리 내에서 명령어 : make

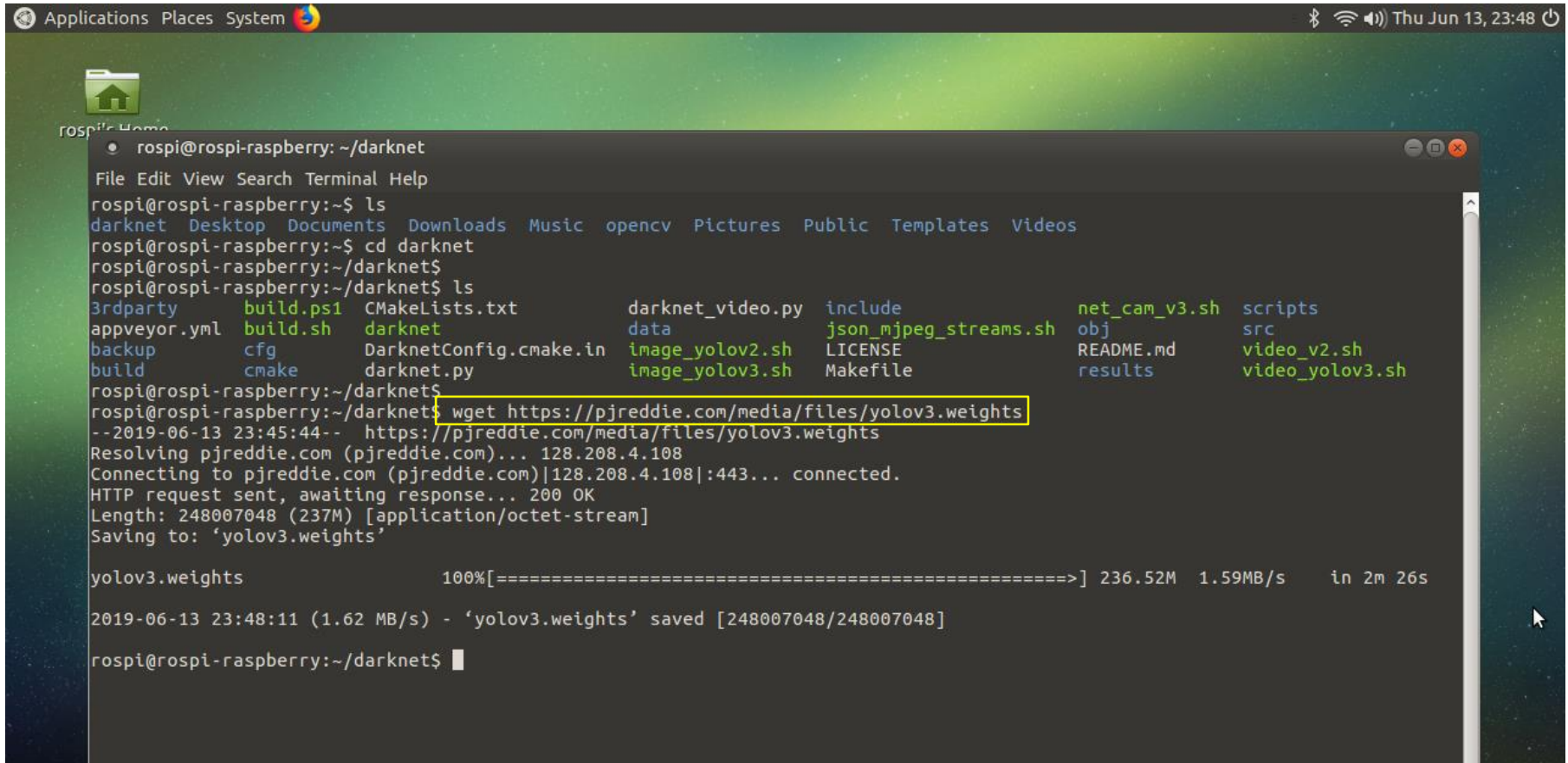


```
rospi@rospi-raspberry: ~/darknet
File Edit View Search Terminal Help
rospi@rospi-raspberry:~$ ls
darknet Desktop Documents Downloads Music opencv Pictures Public Templates Videos
rospi@rospi-raspberry:~$
rospi@rospi-raspberry:~$ cd darknet
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ ls
3rdparty      build.sh      DarknetConfig.cmake.in  image_yolov2.sh  LICENSE      scripts
appveyor.yml  cfg           darknet.py              image_yolov3.sh  Makefile     src
build         cmake         darknet_video.py        include          net_cam_v3.sh video_v2.sh
build.ps1     CMakeLists.txt data                  json_mjpeg_streams.sh  README.md    video_yolov3.sh
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ sudo nano Makefile
[sudo] password for rospi:
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ make
```

Run YOLO v3

■ YOLO v3 Weight file Download

- Darknet 디렉토리 내에서 명령어 : `wget https://pjreddie.com/media/files/yolov3.weights`



The screenshot shows a terminal window on a Raspberry Pi. The user is in the `~/darknet` directory. They run `ls` to list files, then `cd darknet`. They run `ls` again, showing a list of files and directories. Then they run `wget https://pjreddie.com/media/files/yolov3.weights`, which is highlighted with a yellow box. The terminal shows the progress of the download, including the URL, the file size (236.52M), the download speed (1.59MB/s), and the time taken (2m 26s). The download is successful, and the file is saved to the current directory.

```
rospi@rospi-raspberry: ~/darknet
File Edit View Search Terminal Help
rospi@rospi-raspberry:~$ ls
darknet Desktop Documents Downloads Music opencv Pictures Public Templates Videos
rospi@rospi-raspberry:~$ cd darknet
rospi@rospi-raspberry:~/darknet$ ls
3rdparty      build.ps1    CMakeLists.txt      darknet_video.py  include        net_cam_v3.sh  scripts
appveyor.yml  build.sh     darknet              data              json_mjpeg_streams.sh  obj            src
backup        cfg          DarknetConfig.cmake.in  image_yolov2.sh  LICENSE        README.md      video_v2.sh
build         cmake        darknet.py           image_yolov3.sh  Makefile       results        video_yolov3.sh
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ wget https://pjreddie.com/media/files/yolov3.weights
--2019-06-13 23:45:44-- https://pjreddie.com/media/files/yolov3.weights
Resolving pjreddie.com (pjreddie.com)... 128.208.4.108
Connecting to pjreddie.com (pjreddie.com)|128.208.4.108|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 248007048 (237M) [application/octet-stream]
Saving to: 'yolov3.weights'

yolov3.weights          100%[=====>] 236.52M  1.59MB/s   in 2m 26s

2019-06-13 23:48:11 (1.62 MB/s) - 'yolov3.weights' saved [248007048/248007048]

rospi@rospi-raspberry:~/darknet$
```


Run YOLO v3

■ YOLO v3 단일 이미지 Object-Detection 실행

- 명령어 : `./darknet detect cfg/yolov3.cfg yolov3.weights data/dog.jpg`

```
rospi@rospi-raspberry: ~/darknet
File Edit View Search Terminal Help
rospi@rospi-raspberry:~$ ls
darknet Desktop Documents Downloads Music opencv Pictures Public Templates Videos
rospi@rospi-raspberry:~$ cd darknet
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ ls
3rdparty      build.ps1    CMakeLists.txt  darknet_video.py  include      net_cam_v3.sh  scripts
appveyor.yml  build.sh     darknet          data              json_mjpeg_streams.sh  obj            src
backup        cfg          DarknetConfig.cmake.in  image_yolov2.sh  LICENSE      README.md      video_v2.sh
build         cmake        darknet.py       image_yolov3.sh  Makefile     results        video_yolov3.sh
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ wget https://pjreddie.com/media/files/yolov3.weights
--2019-06-13 23:45:44-- https://pjreddie.com/media/files/yolov3.weights
Resolving pjreddie.com (pjreddie.com)... 128.208.4.108
Connecting to pjreddie.com (pjreddie.com)|128.208.4.108|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 248007048 (237M) [application/octet-stream]
Saving to: 'yolov3.weights'

yolov3.weights          100%[=====] 236.52M  1.59MB/s   in 2m 26s
2019-06-13 23:48:11 (1.62 MB/s) - 'yolov3.weights' saved [248007048/248007048]

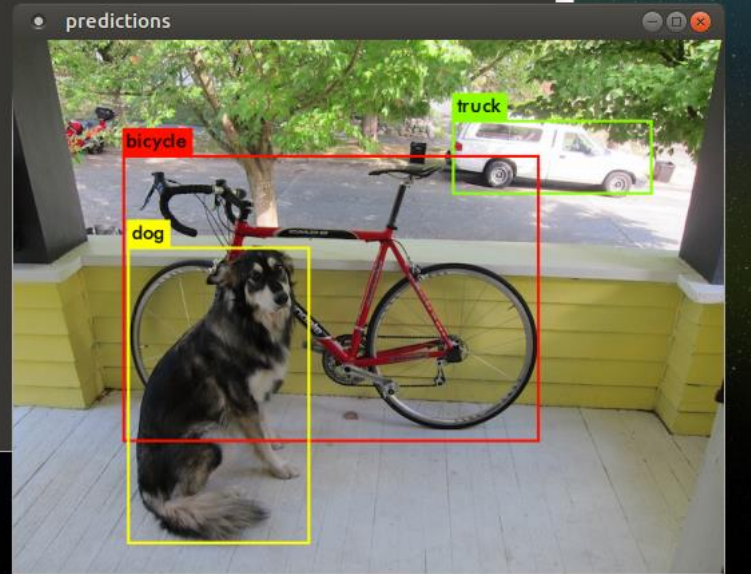
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ ls
3rdparty      build.sh     DarknetConfig.cmake.in  image_yolov3.sh  net_cam_v3.sh  src
appveyor.yml  cfg         darknet.py              include          obj            video_v2.sh
backup        cmake       darknet_video.py       json_mjpeg_streams.sh  README.md      video_yolov3.sh
build         CMakeLists.txt  data                  LICENSE          results        yolov3.weights
build.ps1     darknet      image_yolov2.sh        Makefile         scripts
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$
rospi@rospi-raspberry:~/darknet$ ./darknet detect cfg/yolov3.cfg yolov3.weights data/dog.jpg
```

Run YOLO v3

■ YOLO v3 단일 이미지 Object-Detection 실행

- Processing 완료 및 Object Detection 여부 확인

```
89 conv 256 1 x 1 / 1(1) 26 x 26 x 512 -> 26 x 26 x 256 0.177 BF
90 conv 512 3 x 3 / 1(1) 26 x 26 x 256 -> 26 x 26 x 512 1.595 BF
91 conv 256 1 x 1 / 1(1) 26 x 26 x 512 -> 26 x 26 x 256 0.177 BF
92 conv 512 3 x 3 / 1(1) 26 x 26 x 256 -> 26 x 26 x 512 1.595 BF
93 conv 255 1 x 1 / 1(1) 26 x 26 x 512 -> 26 x 26 x 255 0.177 BF
94 yolo
[yolo] params: iou loss: mse, iou_norm: 0.75, cls_norm: 1.00, scale_x_y: 1.00
95 route 91
96 conv 128 1 x 1 / 1(1) 26 x 26 x 256 -> 26 x 26 x 128 0.044 BF
97 upsample 2x 26 x 26 x 128 -> 52 x 52 x 128
98 route 97 36
99 conv 128 1 x 1 / 1(1) 52 x 52 x 384 -> 52 x 52 x 128 0.266 BF
100 conv 256 3 x 3 / 1(1) 52 x 52 x 128 -> 52 x 52 x 256 1.595 BF
101 conv 128 1 x 1 / 1(1) 52 x 52 x 256 -> 52 x 52 x 128 0.177 BF
102 conv 256 3 x 3 / 1(1) 52 x 52 x 128 -> 52 x 52 x 256 1.595 BF
103 conv 128 1 x 1 / 1(1) 52 x 52 x 256 -> 52 x 52 x 128 0.177 BF
104 conv 256 3 x 3 / 1(1) 52 x 52 x 128 -> 52 x 52 x 256 1.595 BF
105 conv 255 1 x 1 / 1(1) 52 x 52 x 256 -> 52 x 52 x 255 0.353 BF
106 yolo
[yolo] params: iou loss: mse, iou_norm: 0.75, cls_norm: 1.00, scale_x_y: 1.00
Total BFLOPS 65.864
Loading weights from yolov3.weights...
seen 64
Done!
data/dog.jpg: Predicted in 698240.288000 milli-seconds.
bicycle: 99%
dog: 100%
truck: 93%
```



■ YOLO_v3 PI Camera를 이용한 Object Detection 실행

- 명령어 : ./darknet detector demo cfg/coco.data cfg/yolov3.cfg yolov3.weights -c 0