

Install OpenCV-Python in Ubuntu

Goals

In this tutorial We will learn to setup OpenCV-Python in Ubuntu System. Below steps are tested for Ubuntu 16.04 and 18.04 (both 64-bit).

OpenCV-Python can be installed in Ubuntu in two ways:

- Install from pre-built binaries available in Ubuntu repositories
- Compile from the source. In this section, we will see both.

Another important thing is the additional libraries required. OpenCV-Python requires only **Numpy** (in addition to other dependencies, which we will see later). But in this tutorials, we also use **Matplotlib** for some easy and nice plotting purposes (which I feel much better compared to OpenCV). Matplotlib is optional, but highly recommended. Similarly we will also see **IPython**, an Interactive Python Terminal, which is also highly recommended.

Installing OpenCV-Python from Pre-built Binaries

This method serves best when using just for programming and developing OpenCV applications.

Install package `python3-opencv` with following command in terminal (as root user).

```
$ sudo apt-get install python3-opencv
```

Open Python IDLE (or IPython) and type following codes in Python terminal.

```
import cv2 as cv
print(cv.__version__)
```

If the results are printed out without any errors, congratulations !!! You have installed OpenCV-Python successfully.

It is quite easy. But there is a problem with this. Apt repositories may not contain the latest version of OpenCV always. For example, at the time of writing this tutorial, apt repository contains 2.4.8 while latest OpenCV version is 3.x. With respect to Python API, latest version will always contain much better support and latest bug fixes.

So for getting latest source codes preference is next method, i.e. compiling from source. Also at some point in time, if you want to contribute to OpenCV, you will need this.

Building OpenCV from source

Compiling from source may seem a little complicated at first, but once you succeeded in it, there is nothing complicated.

First we will install some dependencies. Some are required, some are optional. You can skip optional dependencies if you don't want.

Required build dependencies

We need **CMake** to configure the installation, **GCC** for compilation, **Python-devel** and **Numpy** for building Python bindings etc.

```
sudo apt-get install cmake
sudo apt-get install gcc g++
```

to support python2:

```
sudo apt-get install python-dev python-numpy
```

to support python3:

```
sudo apt-get install python3-dev python3-numpy
```

Next we need **GTK** support for GUI features, Camera support (v4l), Media Support (ffmpeg, gstreamer) etc.

```
sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev
sudo apt-get install libgstreamer-plugins-base1.0-dev libgstreamer1.0-dev
```

to support gtk2:

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

to support gtk3:

```
sudo apt-get install libgtk-3-dev
```

Optional Dependencies

Above dependencies are sufficient to install OpenCV in your Ubuntu machine. But depending upon your requirements, you may need some extra dependencies. A list of such optional dependencies are given below. You can either leave it or install it, your call :)

OpenCV comes with supporting files for image formats like PNG, JPEG, JPEG2000, TIFF, WebP etc. But it may be a little old. If you want to get latest libraries, you can install development files for system libraries of these formats.

```
sudo apt-get install libpng-dev
sudo apt-get install libjpeg-dev
sudo apt-get install libopenexr-dev
sudo apt-get install libtiff-dev
sudo apt-get install libwebp-dev
```

Note

If you are using Ubuntu 16.04 you can also install `libjasper-dev` to add a system level support for the JPEG2000 format.

Downloading OpenCV

To download the latest source from OpenCV's [GitHub Repository](#). (If you want to contribute to OpenCV choose this. For that, you need to install **Git** first)

```
$ sudo apt-get install git
$ git clone https://github.com/opencv/opencv.git
```

It will create a folder "opencv" in current directory. The cloning may take some time depending upon your internet connection.

Now open a terminal window and navigate to the downloaded "opencv" folder. Create a new "build" folder and navigate to it.

```
$ mkdir build
$ cd build
```

Configuring and Installing

Now we have all the required dependencies, let's install OpenCV. Installation has to be configured with CMake. It specifies which modules are to be installed, installation path, which additional libraries to be used, whether documentation and examples to be compiled etc. Most of this work are done automatically with well configured default parameters.

Below command is normally used for configuration of OpenCV library build (executed from build folder):

```
$ cmake ../
```

OpenCV defaults assume "Release" build type and installation path is "/usr/local". For additional information about CMake options refer to OpenCV [C++ compilation guide](#):

You should see these lines in your CMake output (they mean that Python is properly found):

```
-- Python 2:
-- Interpreter:          /usr/bin/python2.7 (ver 2.7.6)
-- Libraries:            /usr/lib/x86_64-linux-gnu/libpython2.7.so (ver 2.7.6)
-- numpy:                /usr/lib/python2.7/dist-packages/numpy/core/include (ver 1.8.2)
-- packages path:        lib/python2.7/dist-packages
--
-- Python 3:
-- Interpreter:          /usr/bin/python3.4 (ver 3.4.3)
-- Libraries:            /usr/lib/x86_64-linux-gnu/libpython3.4m.so (ver 3.4.3)
-- numpy:                /usr/lib/python3/dist-packages/numpy/core/include (ver 1.8.2)
-- packages path:        lib/python3.4/dist-packages
```

Now you build the files using "make" command and install it using "make install" command.

```
$ make
# sudo make install
```

Installation is over. All files are installed in "/usr/local/" folder. Open a terminal and try import "cv2".

```
import cv2 as cv
print(cv.__version__)
```