[dlib](#) is a library developed by [Davis King](#), the dlib C++ library is a cross-platform package for threading, networking, numerical operations, machine learning, computer vision, and compression, placing a strong emphasis on extremely high-quality and portable code. The documentation for the library is also fantastic.

Step 1 - Set up your Python virtual environment

$ workon py3cv3

Step 2 – Install the required Python packages

All pre-requisites required to compile dlib should already be installed to your py3cv3 environment. If not, let pip install them:

$ pip install numpy
$ pip install scipy
$ pip install cmake
$ pip install -U scikit-image

Step 3 (Option A) – PIP

The simplest way to install dlib is with PIP.

To install dlib via PIP, simply enter the following command in your environment:

$ pip install dlib

You can now safely skip to Step 4, but just be aware that imglab isn't available.

Step 3 (Option B) – Compile and install dlib

Unlike OpenCV, the dlib library with Python bindings is very easy to install. The first step is to head to the dlib homepage and grab the latest release of the library. At the time of this writing, the latest version of dlib is 19.10 and can be downloaded using the following link:

http://dlib.net/files/dlib-19.10.tar.bz2

...or using wget as shown below.

Once dlib has been downloaded, we need to unpack it and setup the build directory:

$ wget http://dlib.net/files/dlib-19.10.tar.bz2
$ tar xvjf dlib-19.10.tar.bz2
$ cd dlib-19.10/python_examples/
$ for req in $(cat requirements.txt); do pip install $req; done

And then we can configure and start the compile.

$ cd ..
$ python setup.py install

This will kick off the compile.

At this point, our dlib Python bindings are installed on our system.

Step 4 – Test out dlib installation

```
$ python
>>> import dlib
>>>
```

And sure enough, our dlib package is able to import without an issue.