



CPNV - Centre Professionnel du Nord Vaudois

MCT - Modules complémentaires techniques

Installation OpenCV

P2213

Rédacteur : Quentin Surdez

Relecture : Rafael Dousse

École : CPNV

Date : Yverdon-Les-Bains, le 3 juin 2022



Table des matières

1	Introduction	1
2	Installation	2
3	Conclusion	5



1 Introduction



OpenCV est un logiciel open-source permettant un traitement de l'image. Nous l'utilisons pour reconnaître des codes QR, reconnaître une ligne et encore streamer le flux d'images sur une page Web. Nous l'avons installé en entier, ce que ne permet pas de faire un *pip install*. Pour installer OpenCV en entier, nous avons besoin d'utiliser CMake. Les différentes étapes sont détaillées dans ce document.



2 Installation

La première chose à checker est la version de l'OS utilisée. Nous avons besoin de Bullseye 32bits. Ensuite, nous pouvons lancer l'installation. Je n'ai pas réussi à installer via pip. J'ai uniquement réussi l'installation depuis la source. C'est cette installation que je vais expliquer pas à pas dans ce document.

La première chose à faire est de s'assurer que le système est à jour via ces deux commandes :

```
1      sudo apt update
2      sudo apt upgrade
```

Nous devons maintenant agrandir le filesystem. Pour ce faire, il vous faut taper :

```
3      sudo raspi-config
```

puis aller dans les options avancées et la première option est d'agrandir le filesystem. Il vous faudra reboot le Raspberry PI pour que ce changement soit effectif.

Nous devons ensuite construire l'environnement avec des librairies qui nous permettront d'utiliser OpenCV à sa pleine capacité avec notre Raspberry PI. Chaque batch de librairie sera expliqué.

Nous allons premièrement installé des developper tools, dont CMake, qui vont nous aider à build OpenCV :

```
4      sudo apt-get install build-essential cmake pkg-config
```

Nous allons maintenant télécharger des packages I/O permettant de gérer plusieurs formats d'images. Ces fichiers seront supportés : JPEG, PNG, TIFF, etc.

```
5      sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng-dev
```

Nous avons maintenant les packages qui gèrent les images, nous en avons aussi besoin pour gérer les vidéos :

```
6      sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev
7      sudo apt-get install libxvidcore-dev libx264-dev
```

La librairie OpenCV vient avec un module appelé highgui qui est un module permettant d'afficher les images et les vidéos sur l'espace de travail. Nous allons donc devoir installé toutes les libairies GTK :

```
8      sudo apt install libfontconfig1-dev libcairo2-dev
9      sudo apt install libgdk-pixbuf2.0-dev libpango1.0-dev
10     sudo apt install libgtk2.0-dev libgtk-3-dev
```



Beaucoup d'opérations dans OpenCV utilisent des matrices nous pouvons optimiser ces opérations avec quelques bibliothèques, ces bibliothèques sont vraiment importantes pour des systèmes avec des contraintes de ressources comme notre Raspberry PI :

```
11      sudo apt-get install libatlas-base-dev gfortran
```

Nous devons ensuite installer des bibliothèques HD5 et QT GUIs pour faciliter l'installation, si vous travaillez avec une version Legacy de PI OS il vous faudra les versions des bibliothèques qui correspondent à votre OS :

```
12      sudo apt install libhdf5-dev libhdf5-serial-dev libhdf5-103
13      sudo apt install libqt5gui5 libqt5webkit5 libqt5test5 python3-pyqt5
```

Nous nous assurons que python3 est bien installé sur notre système et téléchargeons numpy pour python3 :

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

Et nous pouvons enfin télécharger OpenCV depuis git, la première chose à faire est donc de s'assurer que git est bien installé sur notre système et ensuite nous téléchargeons OpenCV :

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

Cela va créer un directory opencv dans le home directory. Maintenant, nous devons créer un directory build dans le directory opencv :

```
17      cd ~/opencv
18      mkdir build
19      cd build
```

Et nous pouvons enfin installer OpenCV, attention la commande suivante doit être exécutée depuis le build folder :

```
20      cmake ../
```

Nous pouvons vérifier que l'installation s'est bien déroulée en checkant l'output pour Python3 :

```
—      Python 3:
—      Interpreter: /usr/bin/python3.9 (ver 3.9.x)
—      Libraries: /usr/lib/x86_64-linux-gnu/libpython3.4m.so
—      numpy: /usr/lib/python3/dist-packages/numpy/core/include
—      packages path: lib/python3.4/dist-packages
```

Pour que les prochains processus se fassent rapidement en utilisant toutes les ressources du Raspberry PI. Nous changeons la valeur des SWAPFILES. Cette valeur devra être remise à zéro à la fin de l'installation.

```
21      sudo nano /etc/dphys-swapfile
```



Éditer les fichiers directement :

```
22      # set size to absolute value, leaving empty (default) then uses computed value
23      # you most likely don't want this, unless you have an special disk situation
24      # CONF_SWAPSIZE=100
25      CONF_SWAPSIZE=2048
```

Nous devons restart le swap service :

```
26      sudo /etc/init.d/dphys-swapfile stop
27      sudo /etc/init.d/dphys-swapfile start
```

Et maintenant nous buildons les files pour pouvoir utiliser OpenCV, nous précisons aussi le nombre de core à utiliser pour accélérer le processus :

```
28      make -j4
29      sudo make install
```

La commande make peut prendre entre 2-4 heures.

Nous pouvons vérifier que l'installation s'est bien déroulée en tapant les commandes suivantes dans le terminal :

```
30      python
31      >>> import cv2
32      >>> cv2.__version__
```

Si l'installation s'est bien déroulée l'output de la commande vous donnera la version installée. (4.5.x)



3 Conclusion

Le téléchargement de logiciel par la source est compliquée et requiert beaucoup de temps. Chaque étape doit être suivie scrupuleusement pour être certain d'avoir un output propre et exécutable.