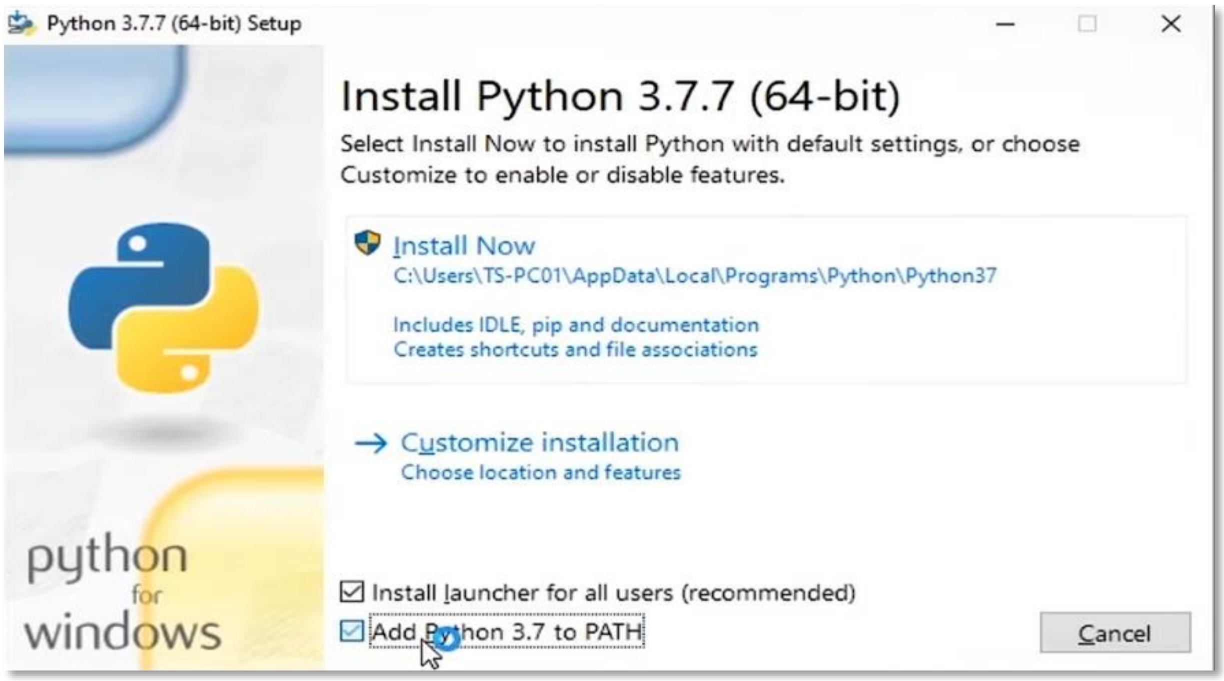
1. Installing Python 3.7

I'm using version 3.7.7, but any version of Python 3.7 will work.
When installing, remember to check the "Add Python to path" box.



Python 3.7.7 (64-bit) Setup

Install Python 3.7.7 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Users\TS-PC01\AppData\Local\Programs\Python\Python37

Includes IDLE, pip and documentation
Creates shortcuts and file associations

→ Customize installation
Choose location and features

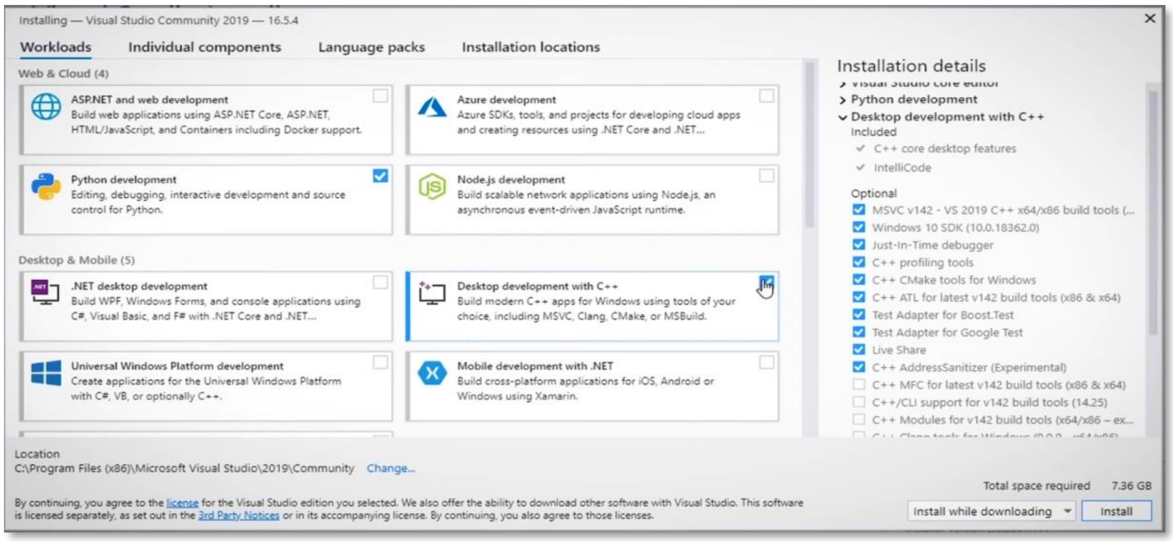
☑ Install launcher for all users (recommended)
☑ Add Python 3.7 to PATH

Cancel

Once installed, go to the console and install Numpy with "pip install numpy".

2. Installing the compilation tools

Download the latest version of CMake and install it. During the installation, leave all default settings.
Download Visual Studio Community 2019. Wait until the installation is complete and on the development tools screen, select "Python development" and "Desktop development with C++".



Installing — Visual Studio Community 2019 — 16.5.4

Workloads | Individual components | Language packs | Installation locations

Web & Cloud (4)

☐ ASP.NET and web development
Build web applications using ASP.NET Core, ASP.NET, HTML/JavaScript, and Containers including Docker support.

☐ Azure development
Azure SDKs, tools, and projects for developing cloud apps and creating resources using .NET Core and .NET...

☑ Python development
Editing, debugging, interactive development and source control for Python.

☐ Node.js development
Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.

Desktop & Mobile (5)

☐ .NET desktop development
Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET Core and .NET...

☑ Desktop development with C++
Build modern C++ apps for Windows using tools of your choice, including MSVC, Clang, CMake, or MSBuild.

☐ Universal Windows Platform development
Create applications for the Universal Windows Platform with C#, VB, or optionally C++.

☐ Mobile development with .NET
Build cross-platform applications for iOS, Android or Windows using Xamarin.

Location
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community Change...

By continuing, you agree to the license for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the 3rd Party Notices or in its accompanying license. By continuing, you also agree to those licenses.

Installation details
> Visual Studio core editor
> Python development
⌄ Desktop development with C++
Included
✓ C++ core desktop features
✓ IntelliCode
Optional
☑ MSVC v142 - VS 2019 C++ x64/x86 build tools (...
☑ Windows 10 SDK (10.0.18362.0)
☑ Just-In-Time debugger
☑ C++ profiling tools
☑ C++ CMake tools for Windows
☑ C++ ATL for latest v142 build tools (x86 & x64)
☑ Test Adapter for Boost.Test
☑ Test Adapter for Google Test
☑ Live Share
☑ C++ AddressSanitizer (Experimental)
☐ C++ MFC for latest v142 build tools (x86 & x64)
☐ C++/CLI support for v142 build tools (14.25)
☐ C++ Modules for v142 build tools (x64/x86 – ex...

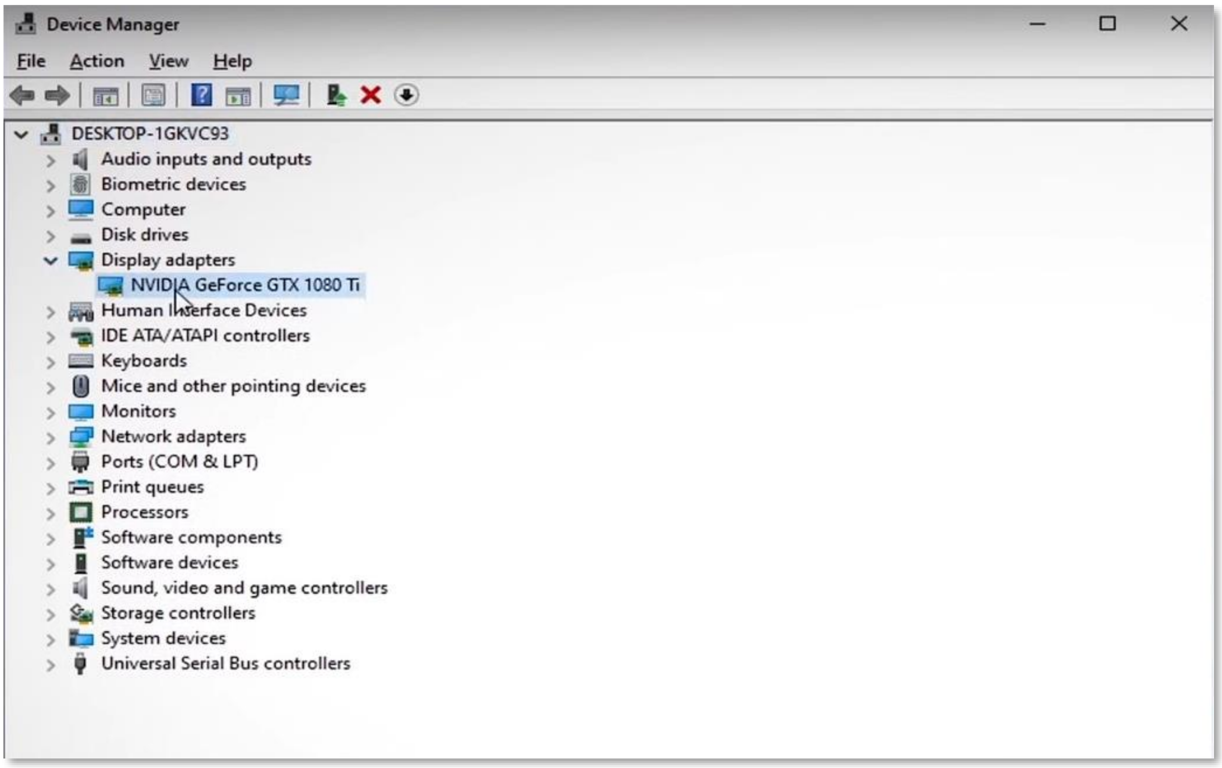
Total space required 7.36 GB

Install while downloading | Install

The installation will take some time. Reboot the computer after the installation is complete.

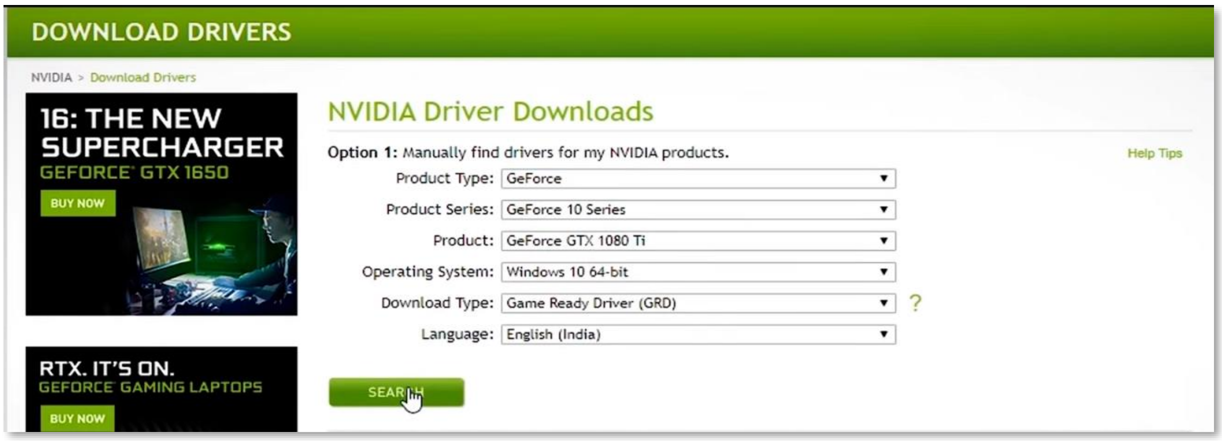
3. Updating Graphics Drivers

In the Windows search bar, type "device manager". Open the "Graphics Cards" tab. You need an NVIDIA graphics card to be able to use the plugin with the GPU.



DESKTOP-1GKVC93
- Audio inputs and outputs
- Biometric devices
- Computer
- Disk drives
- Display adapters
  - NVIDIA GeForce GTX 1080 Ti
- Human Interface Devices
- IDE ATA/ATAPI controllers
- Keyboards
- Mice and other pointing devices
- Monitors
- Network adapters
- Ports (COM & LPT)
- Print queues
- Processors
- Software components
- Software devices
- Sound, video and game controllers
- Storage controllers
- System devices
- Universal Serial Bus controllers

Make a note of the card name and go to the "Download drivers | NVIDIA" web page. Fill in the fields with information specific to the graphics card. Make sure to select "Windows 10 64-bit" and for the Download Type, "Game Driver Ready" is all good. Download and install the driver.



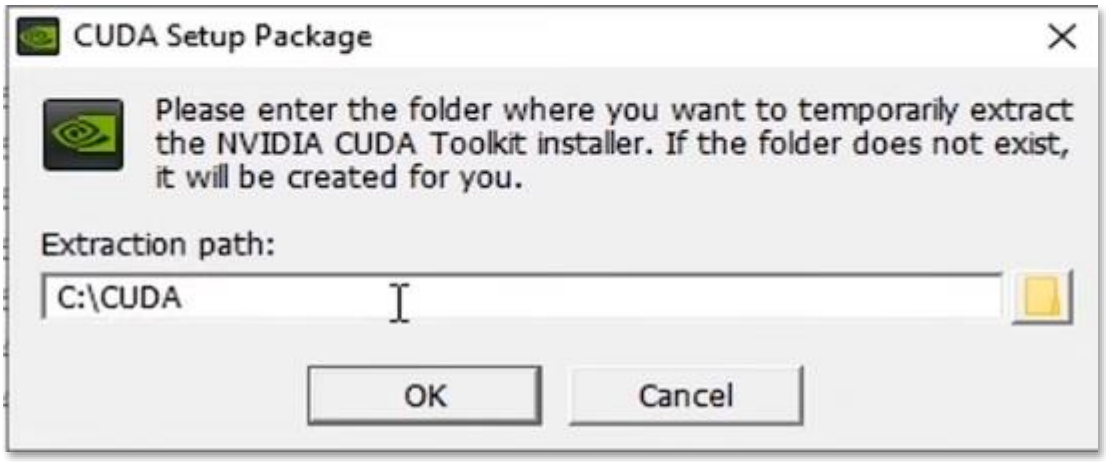
NVIDIA Driver Downloads

Option 1: Manually find drivers for my NVIDIA products.

Product Type:	GeForce
Product Series:	GeForce 10 Series
Product:	GeForce GTX 1080 Ti
Operating System:	Windows 10 64-bit
Download Type:	Game Ready Driver (GRD)
Language:	English (India)

4. CUDA Installation

Check on "CUDA Support Wikipedia" that the graphics card is compatible with CUDA 3.0 at least. Search for "nvidia cuda 10.2", click on the first link, select the corresponding Windows version and "exe (local)". Download CUDA, this will take some time.
Start the installation and change the path extraction to "C:\CUDA".


CUDA Setup Package

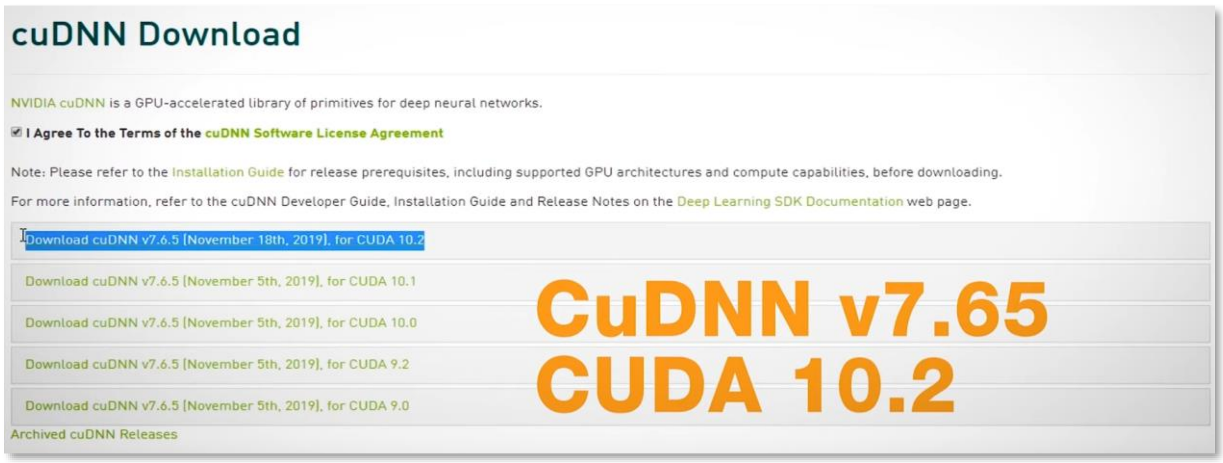
Please enter the folder where you want to temporarily extract the NVIDIA CUDA Toolkit installer. If the folder does not exist, it will be created for you.

Extraction path:
C:\CUDA

In the installer, leave everything as default and let the installation proceed.

5. Installing cuDNN

Type "cudnn download" into Google and go to the "NVIDIA cuDNN | NVIDIA Developer" page. To download cuDNN, you will need to create an NVIDIA Developer account, then return to the page and sign in. Choose "cuDNN v7.6.5 [November 18th, 2019], for CUDA 10.2", and click on "cuDNN Library for Widnows 10".


cuDNN Download

NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.

☑ I Agree To the Terms of the cuDNN Software License Agreement

Note: Please refer to the Installation Guide for release prerequisites, including supported GPU architectures and compute capabilities, before downloading.

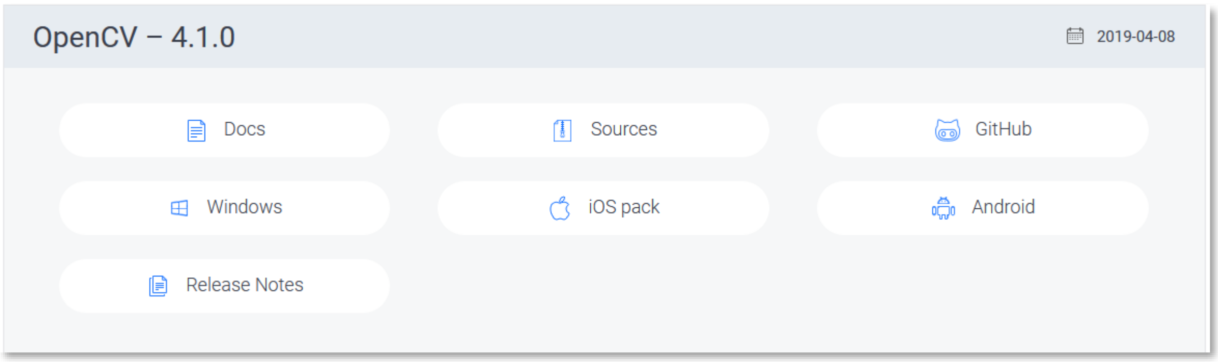
For more information, refer to the cuDNN Developer Guide, Installation Guide and Release Notes on the Deep Learning SDK Documentation web page.

Download cuDNN v7.6.5 (November 18th, 2019), for CUDA 10.2
Download cuDNN v7.6.5 (November 5th, 2019), for CUDA 10.1
Download cuDNN v7.6.5 (November 5th, 2019), for CUDA 10.0
Download cuDNN v7.6.5 (November 5th, 2019), for CUDA 9.2
Download cuDNN v7.6.5 (November 5th, 2019), for CUDA 9.0
Archived cuDNN Releases

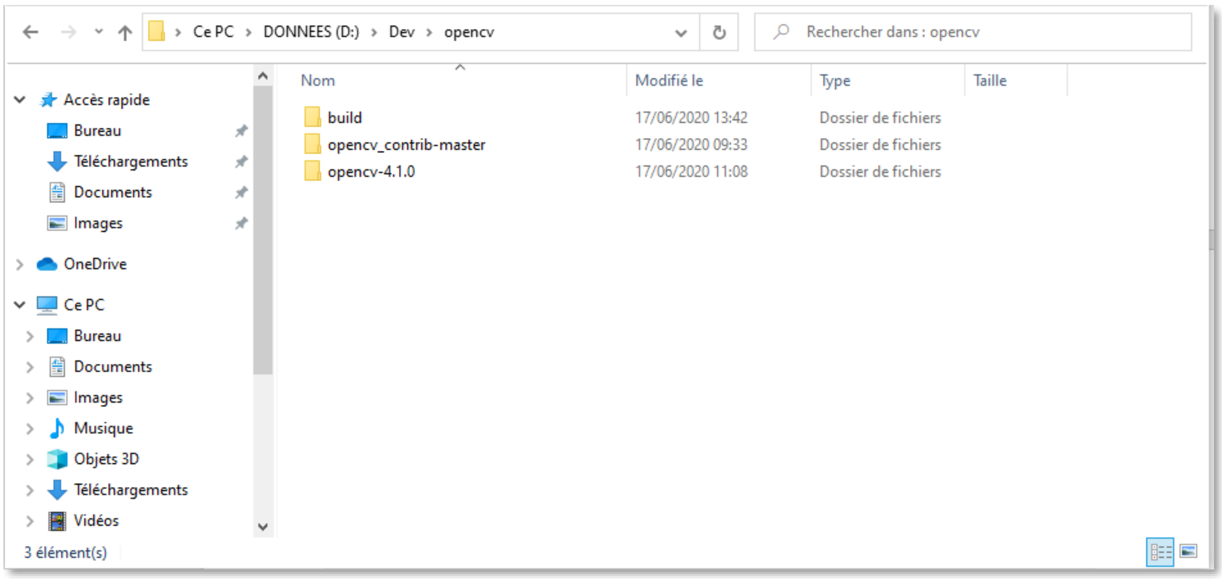
Extract the compressed file.
- In cuda/bin, copy "cudnn64_7.dll" and paste it into "C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\bin".
- In cuda/include, copy "cudnn.h" and paste it into "C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\include".
- In cuda/lib, copy "cudnn.lib" and paste it into "C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\lib\x64".

6. Installing Open CV

Go to "opencv.org", then in the "Release" tab, choose version 4.1.0 and click on "Sources".



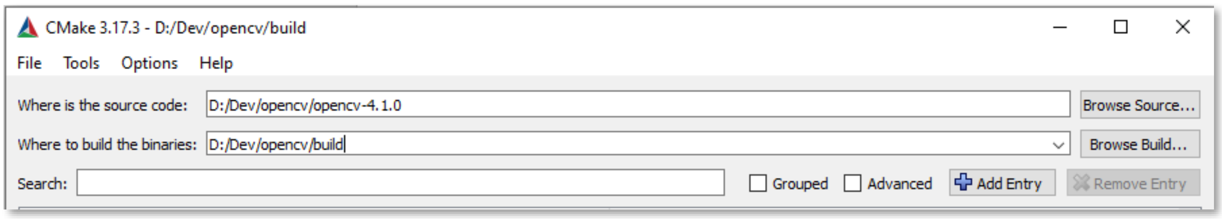
Go to "github.com/opencv/opencv_contrib" and download the zip. On the C disk, create a folder "opencv". Unzip the two files "opencv-4.1.0" and "opencv_contrib-master" and put them in the "opencv" folder. Then create a third folder "build".



Nom	Modifié le	Type	Taille
build	17/06/2020 13:42	Dossier de fichiers	
opencv_contrib-master	17/06/2020 09:33	Dossier de fichiers	
opencv-4.1.0	17/06/2020 11:08	Dossier de fichiers	

7. Configuring CMake

Open CMake and set the source path to the "opencv-4.1.0" folder. For the build path, set the path to the "build" folder created just before.

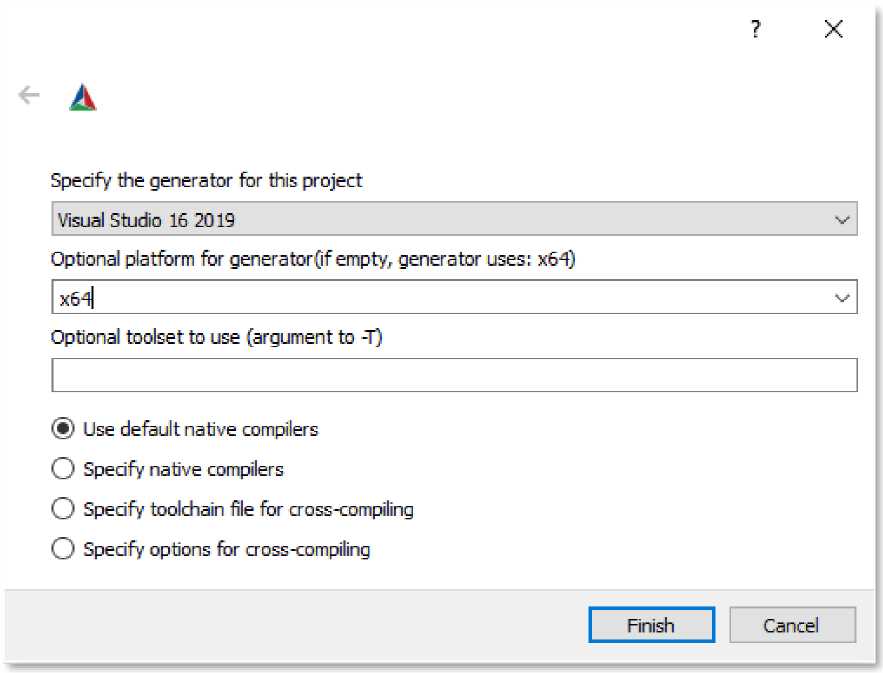


Where is the source code: D:/Dev/opencv/opencv-4.1.0

Where to build the binaries: D:/Dev/opencv/build

Search:

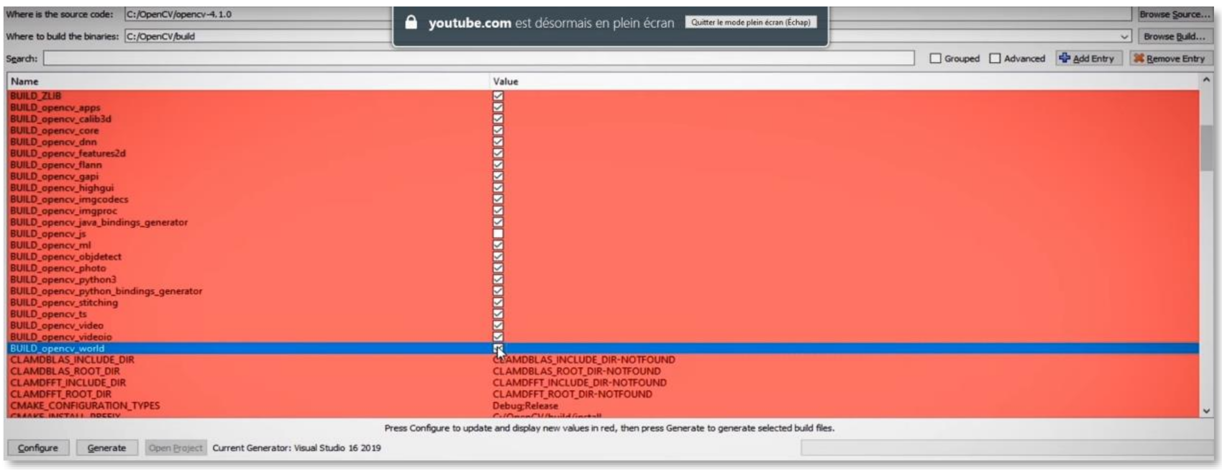
Then click on the "Configure" button and choose "Visual Studio 16 2019" and "x64".



Specify the generator for this project
Visual Studio 16 2019
Optional platform for generator(if empty, generator uses: x64)
x64
Optional toolset to use (argument to -T)

[✓ Use default native compilers]
Specify native compilers
Specify toolchain file for cross-compiling
Specify options for cross-compiling

Click on "Finish". Once the configuration is complete, look in the red modules "BUILD_opencv_world" and check the box on the right. It is better to go check that the version of Python used is the 3.7 installed at the beginning. To do this, look in the "PYTHON3_EXECUTABLE" modules, which will be set to the path of the Python version used by OpenCV. This path can be modified, but you will have to do the same with the paths of the other Python3 modules.



Where is the source code: C:/OpenCV/opencv-4.1.0
Where to build the binaries: C:/OpenCV/build
Search:

Name	Value
BUILD_ZLIB	✓
BUILD_opencv_apps	✓
BUILD_opencv_calib3d	✓
BUILD_opencv_core	✓
BUILD_opencv_dnn	✓
BUILD_opencv_features2d	✓
BUILD_opencv_flann	✓
BUILD_opencv_gapi	✓
BUILD_opencv_highgui	✓
BUILD_opencv_imgcodecs	✓
BUILD_opencv_imgproc	✓
BUILD_opencv_java_bindings_generator	✓
BUILD_opencv_js	☐
BUILD_opencv_ml	✓
BUILD_opencv_objdetect	✓
BUILD_opencv_photo	✓
BUILD_opencv_python3	✓
BUILD_opencv_python_bindings_generator	✓
BUILD_opencv_stitching	✓
BUILD_opencv_ts	✓
BUILD_opencv_video	✓
BUILD_opencv_videoio	✓
BUILD_opencv_world	✓
CLAMDBLAS_INCLUDE_DIR	CLAMDBLAS_INCLUDE_DIR-NOTFOUND
CLAMDBLAS_ROOT_DIR	CLAMDBLAS_ROOT_DIR-NOTFOUND
CLAMDFFT_INCLUDE_DIR	CLAMDFFT_INCLUDE_DIR-NOTFOUND
CLAMDFFT_ROOT_DIR	CLAMDFFT_ROOT_DIR-NOTFOUND
CMAKE_CONFIGURATION_TYPES	Debug;Release

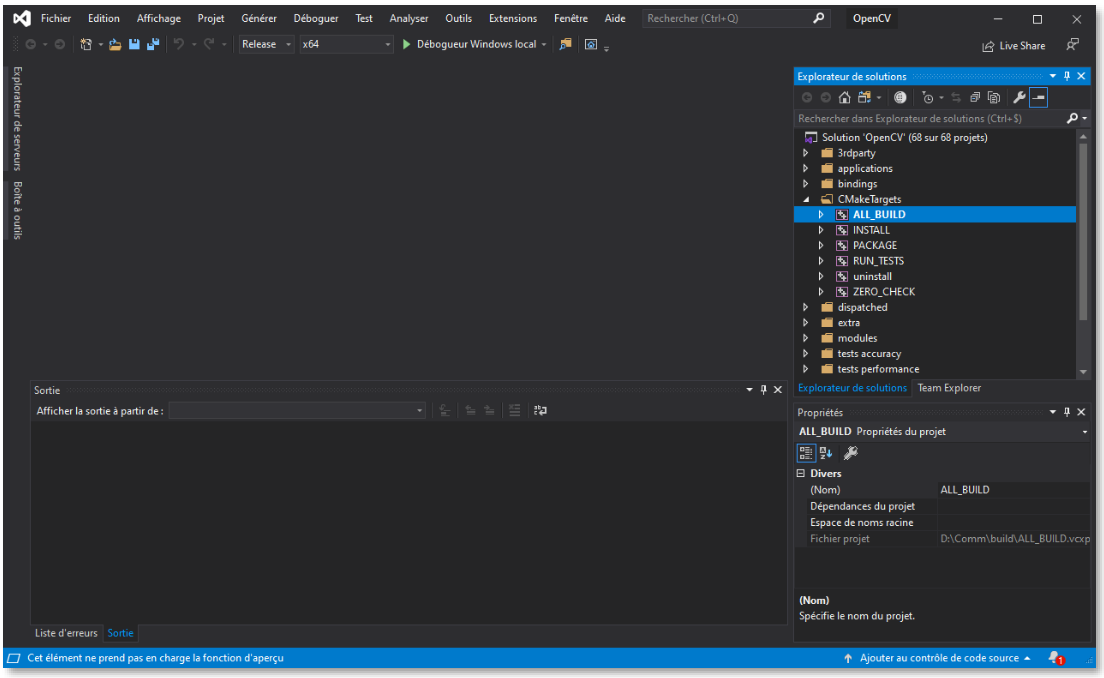
Press Configure to update and display new values in red, then press Generate to generate selected build files.

Current Generator: Visual Studio 16 2019

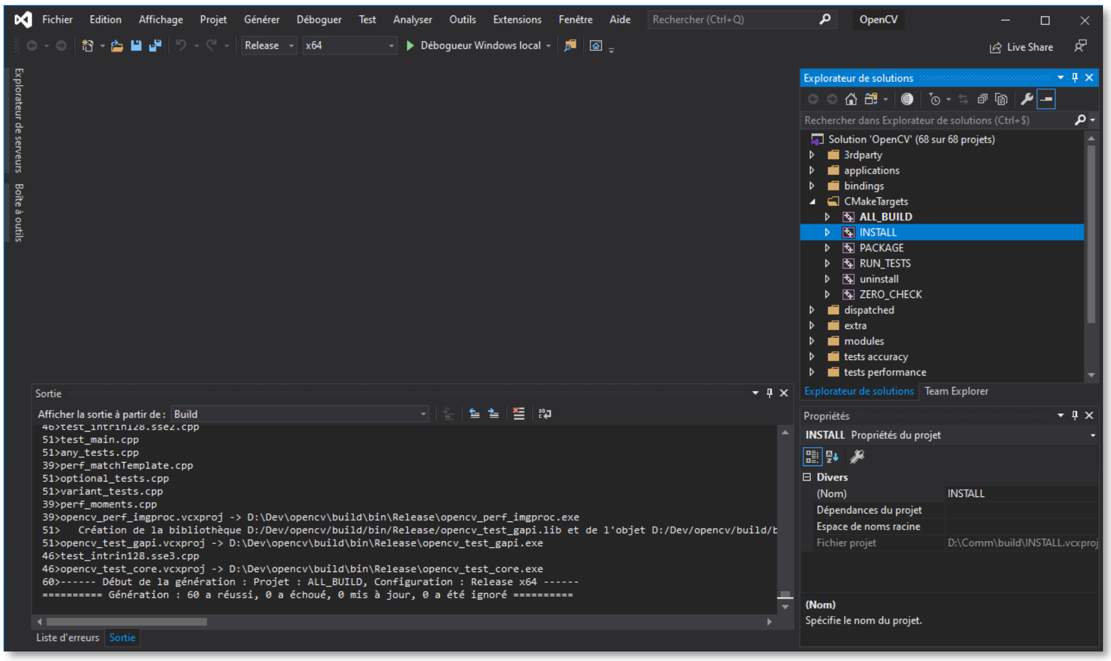
Then click on "Configure" and if all goes well CMake will display "Configuring done, Generating done".

8. Compiling OpenCV with Visual Studio

Go to "opencv/build" and open "ALL_BUILD.vcxproj" with Visual Studio. Be careful, you have to launch Visual Studio as administrator. If it asks to create an account, log in with a Microsoft account or create one.
Put "Release" and "x64" in the menu at the top. In the Solution Explorer, open "CMakeTargets," then right-click on "ALL_BUILD" and choose "Generate.



(Nom) ALL_BUILD
Fichier projet D:\Comm\build\ALL_BUILD.vcxp



```
46>test_intrin128.sse2.cpp
51>test_main.cpp
51>any_tests.cpp
39>perf_matchTemplate.cpp
51>optional_tests.cpp
51>variant_tests.cpp
39>perf_moments.cpp
39>opencv_perf_imgproc.vcxproj -> D:\Dev\opencv\build\bin\Release\opencv_perf_imgproc.exe
51>   Création de la bibliothèque D:/Dev/opencv/build/bin/Release/opencv_test_gapi.lib et de l'objet D:/Dev/opencv/build/b
51>opencv_test_gapi.vcxproj -> D:\Dev\opencv\build\bin\Release\opencv_test_gapi.exe
46>test_intrin128.sse3.cpp
46>opencv_test_core.vcxproj -> D:\Dev\opencv\build\bin\Release\opencv_test_core.exe
60>------ Début de la génération : Projet : ALL_BUILD, Configuration : Release x64 ------
========== Génération : 60 a réussi, 0 a échoué, 0 mis à jour, 0 a été ignoré ==========
```

(Nom) INSTALL
Fichier projet D:\Comm\build\INSTALL.vcxproj

The result should be none failed and 60 successful. Generate the "INSTALL" file as well. The result should be none failed, 2 successful and 58 updated. If you don't have the same number of hits, don't worry as long as there are 0 failures. Check if Open CV is installed, in the console type "python" and type "import cv2". If no error, continue with "print(cv2.__version__)" and check that the version displayed is 4.1.0. If yes, Open CV is now correctly installed and compatible with the GPU of the graphics card.

9. Installing Pupil Capture

On the Pupil Labs website download the Pupil Capture installer. Install the software and open Pupil Capture. A "pupil_capture_settings" folder should be in C:\Users\{your name}\. If you can't find it, change settings in Pupil Capture and check that an eye tracking device is connected to the computer. Starting a recording can help too. If it still doesn't appear, create it by hand.

10. Adding the plugin

In the "pupil_capture_settings" folder, open the plugin folder (or create it if it doesn't exist).
Now paste the folder "darknet", "data", "imagezmq" and the file "detection_plugin.py" into this folder.
If the plugin doesn't appear the next time you open Pupil Capture, look at the "capture.log" file in the "pupil_capture_settings" folder to understand the problem.

11. Classic errors

- When generating OpenCV, make sure that you launch Visual Studio as administrator, then once launched load the project "ALL_BUILD.vcxproj" in the "build" folder.

- If when loading the model in pupil capture, an error appears such as "can't find coco.names" or any other .names file: go to the build/darknet/x64 folder of darknet, then in cfg, open the .names file and in the field "names =" put the absolute path of the .names file, which is in x64/data/coco.names.

- If you get the error "try to set subdivisions to 64", get the .cfg file of the template you load (yolov4 for Yolo 608) and change the value of the subdivisions field to 64. Attention there are commented lines.

- Generally speaking, the darknet files that you will have to modify will be in darknet/build/darknet/x64/.

- If a model doesn't load, you can check in detection_plugin.py, in the load_model() function, the values of configPath, weightPath and metaPath and enter the paths to the corresponding .cfg, .weights and .data files. Be careful with the .data file, you have to take the one in the x64/cfg/ folder.

- Finally to change the color of the boxes, in the plugin file, in the __init__() method, you can change by entering a new BGR value (and not RGB, be careful).