**VPTV software installation guide**

## Contents

# 1 Introduction

This is a step-by-step installation guide on the UIUC ABE VPTV system software installation and compiling. The code is available on the following GitHub page:

https://github.com/m123azizi/LagrangianParticleTracker/

The GitHub documentation is rather old, and with the new software packages, there is a need to properly update the documentation. This guide serves this purpose. The reader should consider:

1. The code on the GitHub page cannot be compiled, and this document is tested on another version of the same code with some functionalities reduced by commenting some code lines. So, this document works on the commented version currently.
2. First, read the GitHub page carefully, and then this document until the end but do not perform the steps in any. Read this document again and perform the steps in the order provided.
3. The GitHub code uses the following packages: QT, VTK, CMake, Boost, YAML CPP, CUDA, OpenCV, OptiTrack Camera SDK, and Libtorch.
4. According to my PC specs, I have updated some of the packages, and some others are unchanged. A short description is given at the beginning of each package guide.
5. This document is accompanied by several screenshots of the installation and compilation process, as well as the working versions of the packages and the commented main code.
6. This document has been tested on Microsoft Windows 11.

Mahdi Azizi

Version 1, Nov 2023

Wood Lab @ ABE @ UIUC

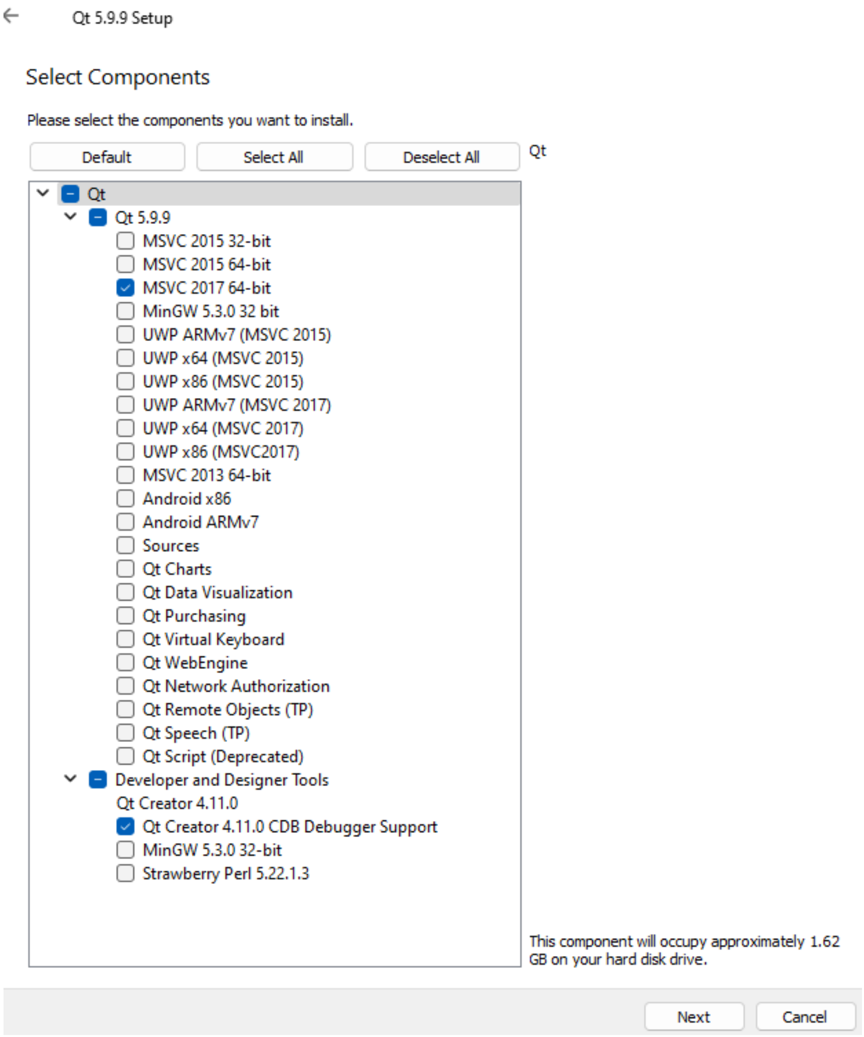
# 2 QT installation

The code uses QT 5.9.9 to build the GUI. Since the old software GUI is fine and currently, I do not intend to change it, this package is not updated. You can install QT by the offline installation file provided ‘qt-opensource-windows-x86-5.9.9.exe’. You can also use the online installer and find QT 5.9.9 from the archive. For both, you will need a free QT account. Make sure to select MSVC 2017 64-bit under the QT branch, as shown below.

I used the offline installer, and my installation directory is C:\Qt\Qt5.9.9\5.9.9\msvc2017_64\bin.

After finishing the installation, add the following to your PC environment variables (see figures):

- Under system variables, add QTDIR pointing to the installation directory.
- Under path in system variables, add ‘%QTDIR%\lib’ And ‘%QTDIR%\bin’



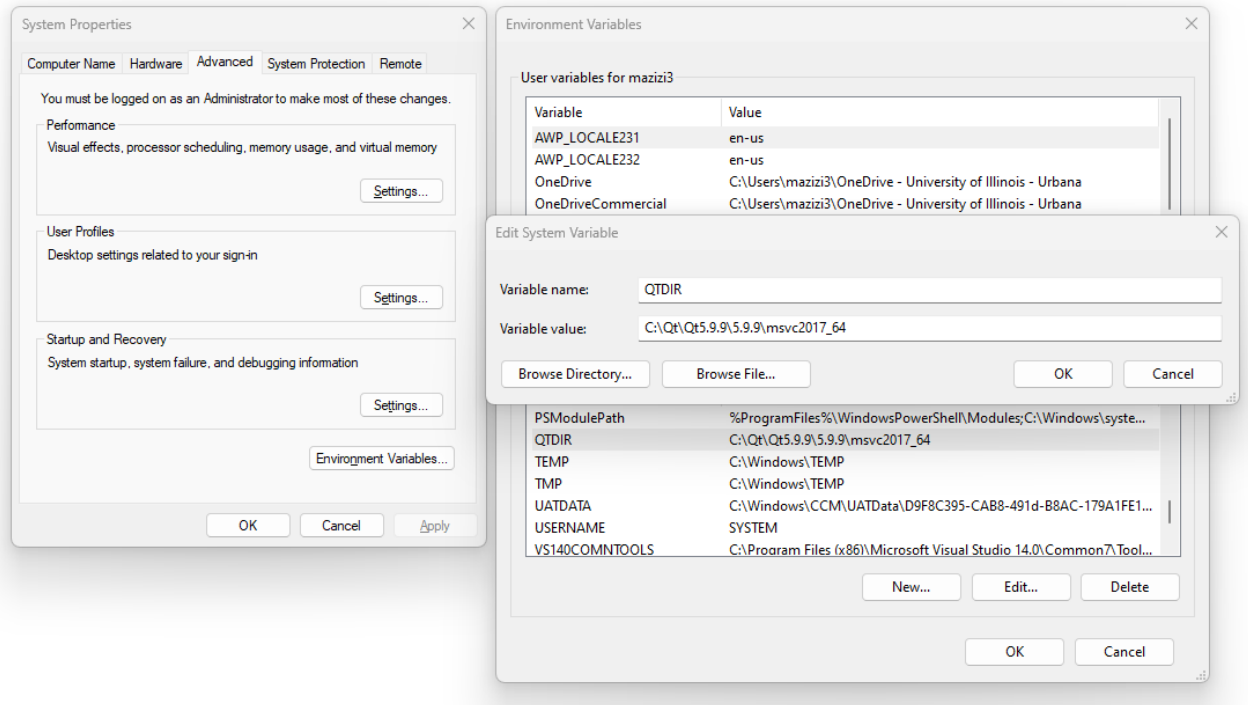
Qt 5.9.9 Setup

Select Components

Please select the components you want to install.

- ■ Qt
  - ■ Qt 5.9.9
    - ☐ MSVC 2015 32-bit
    - ☐ MSVC 2015 64-bit
    - ☑ MSVC 2017 64-bit
    - ☐ MinGW 5.3.0 32 bit
    - ☐ UWP ARMv7 (MSVC 2015)
    - ☐ UWP x64 (MSVC 2015)
    - ☐ UWP x86 (MSVC 2015)
    - ☐ UWP ARMv7 (MSVC 2017)
    - ☐ UWP x64 (MSVC 2017)
    - ☐ UWP x86 (MSVC2017)
    - ☐ MSVC 2013 64-bit
    - ☐ Android x86
    - ☐ Android ARMv7
    - ☐ Sources
    - ☐ Qt Charts
    - ☐ Qt Data Visualization
    - ☐ Qt Purchasing
    - ☐ Qt Virtual Keyboard
    - ☐ Qt WebEngine
    - ☐ Qt Network Authorization
    - ☐ Qt Remote Objects (TP)
    - ☐ Qt Speech (TP)
    - ☐ Qt Script (Deprecated)
  - ■ Developer and Designer Tools
    - Qt Creator 4.11.0
    - ☑ Qt Creator 4.11.0 CDB Debugger Support
    - ☐ MinGW 5.3.0 32-bit
    - ☐ Strawberry Perl 5.22.1.3

This component will occupy approximately 1.62 GB on your hard disk drive.



You must be logged on as an Administrator to make most of these changes.

Performance
Visual effects, processor scheduling, memory usage, and virtual memory

User Profiles
Desktop settings related to your sign-in

Startup and Recovery
System startup, system failure, and debugging information

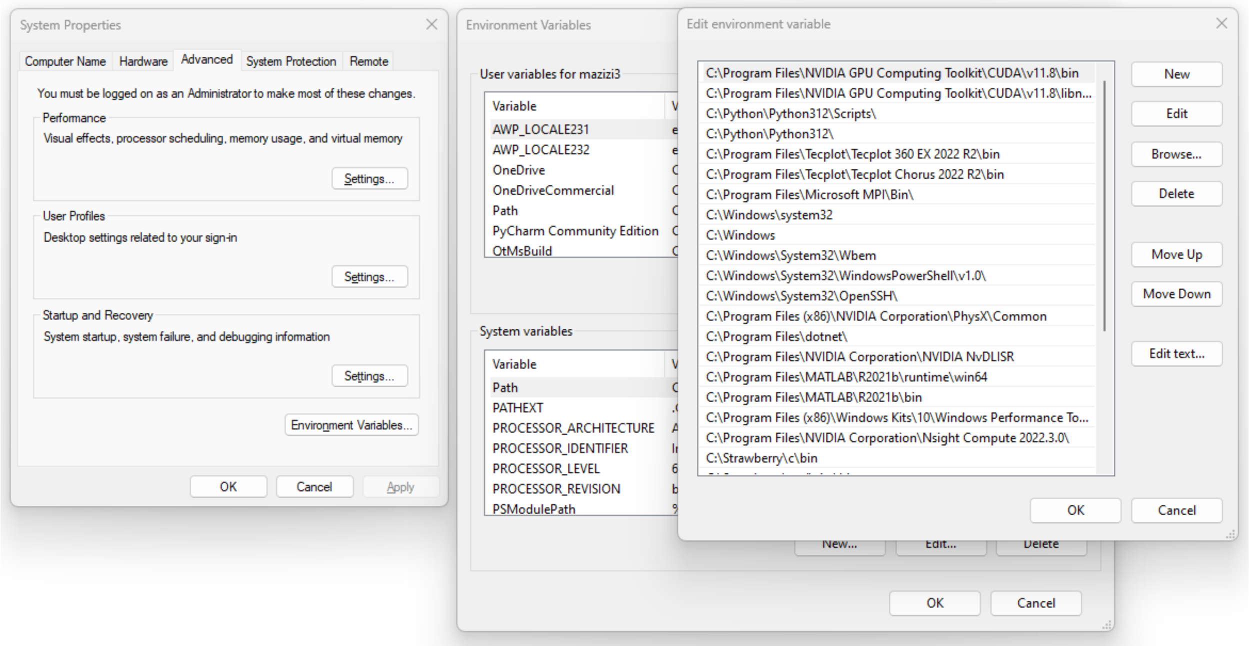
User variables for mazizi3

| Variable | Value |
|---|---|
| AWP_LOCALE231 | en-us |
| AWP_LOCALE232 | en-us |
| OneDrive | C:\Users\mazizi3\OneDrive - University of Illinois - Urbana |
| OneDriveCommercial | C:\Users\mazizi3\OneDrive - University of Illinois - Urbana |

Variable name: QTDIR

Variable value: C:\Qt\Qt5.9.9\5.9.9\msvc2017_64

| | |
|---|---|
| PSModulePath | %ProgramFiles%\WindowsPowerShell\Modules;C:\Windows\syste... |
| QTDIR | C:\Qt\Qt5.9.9\5.9.9\msvc2017_64 |
| TEMP | C:\Windows\TEMP |
| TMP | C:\Windows\TEMP |
| UATDATA | C:\Windows\CCM\UATData\D9F8C395-CAB8-491d-B8AC-179A1FE1... |
| USERNAME | SYSTEM |
| VS140COMNTOOLS | C:\Program Files (x86)\Microsoft Visual Studio 14.0\Common7\Tool... |



You must be logged on as an Administrator to make most of these changes.

Performance
Visual effects, processor scheduling, memory usage, and virtual memory

User Profiles
Desktop settings related to your sign-in

Startup and Recovery
System startup, system failure, and debugging information

User variables for mazizi3

| Variable |
|---|
| AWP_LOCALE231 |
| AWP_LOCALE232 |
| OneDrive |
| OneDriveCommercial |
| Path |
| PyCharm Community Edition |
| OtMsBuild |

System variables

| Variable |
|---|
| Path |
| PATHEXT |
| PROCESSOR_ARCHITECTURE |
| PROCESSOR_IDENTIFIER |
| PROCESSOR_LEVEL |
| PROCESSOR_REVISION |
| PSModulePath |

```
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.8\bin
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.8\libn...
C:\Python\Python312\Scripts\
C:\Python\Python312\
C:\Program Files\Tecplot\Tecplot 360 EX 2022 R2\bin
C:\Program Files\Tecplot\Tecplot Chorus 2022 R2\bin
C:\Program Files\Microsoft MPI\Bin\
C:\Windows\system32
C:\Windows
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
C:\Windows\System32\OpenSSH\
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common
C:\Program Files\dotnet\
C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR
C:\Program Files\MATLAB\R2021b\runtime\win64
C:\Program Files\MATLAB\R2021b\bin
C:\Program Files (x86)\Windows Kits\10\Windows Performance To...
C:\Program Files\NVIDIA Corporation\Nsight Compute 2022.3.0\
C:\Strawberry\c\bin
```

# 3 CUDA installation

You need to have Cuda and Cudnn installed. Follow the guides in the links below, and make sure you download and install the versions for your graphics card. I have Nvidia 4090, so I downloaded and installed Cuda 11.8. For me, cuda 12 also works but I couldn’t compile OpenCV with Cuda 12, so I downgraded the version to 11.8. Check the version, download and install guides at the links below:

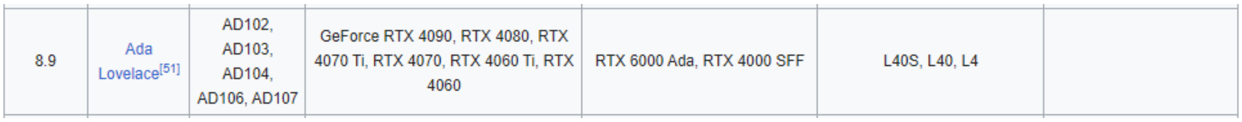
https://en.wikipedia.org/wiki/CUDA

https://medium.com/geekculture/install-cuda-and-cudnn-on-windows-linux-52d1501a8805#3e72

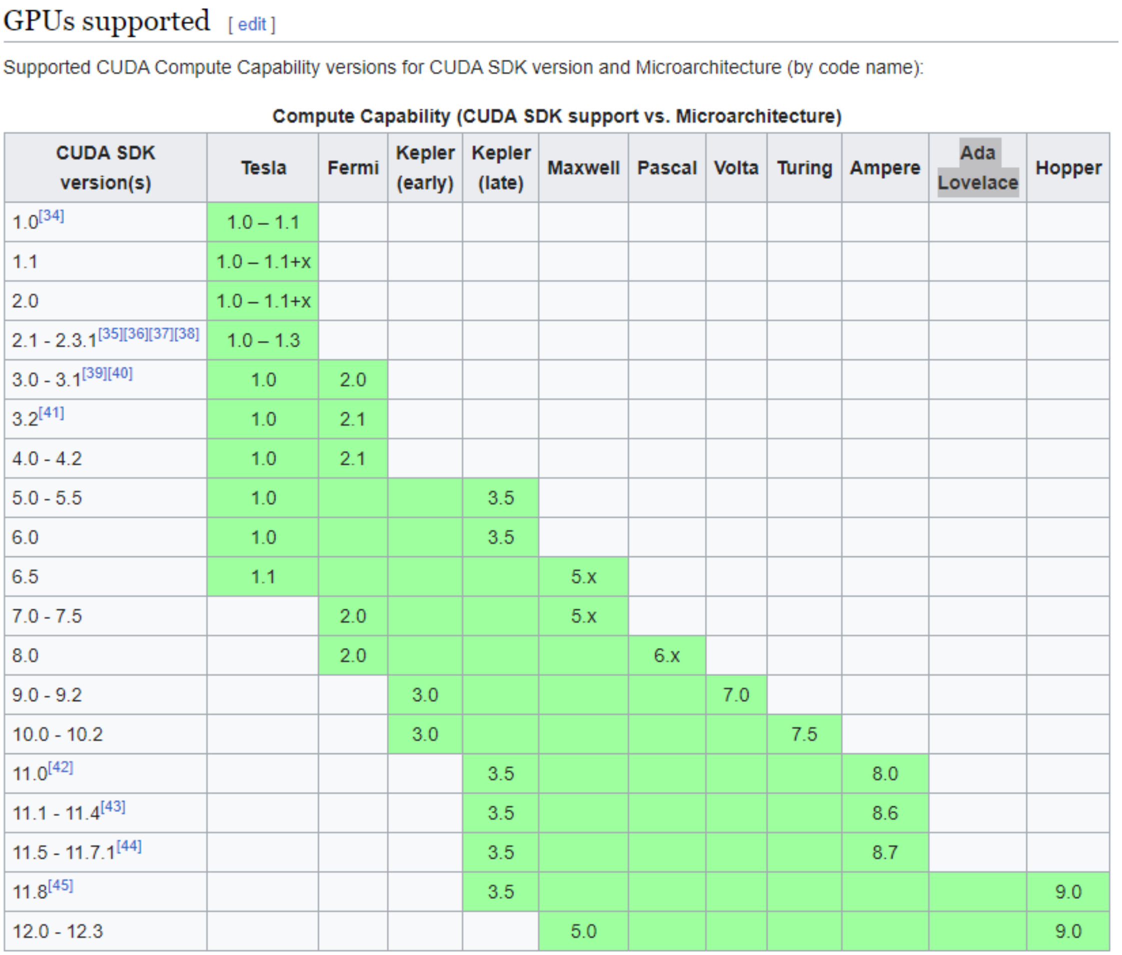
https://docs.nvidia.com/deeplearning/cudnn/install-guide/index.html

For example, I have GeForce RTX 4090, so the Micro-architecture is Ada Lovelace and the Compute capability (architecture binary) is 8.9. We need the architecture binary while building OpenCV in the next steps.

After finishing the installation, make sure Cuda is added to your system environment variables.



| | | | | | | |
|---|---|---|---|---|---|---|
| 8.9 | Ada Lovelace[51] | AD102, AD103, AD104, AD106, AD107 | GeForce RTX 4090, RTX 4080, RTX 4070 Ti, RTX 4070, RTX 4060 Ti, RTX 4060 | RTX 6000 Ada, RTX 4000 SFF | L40S, L40, L4 | |



GPUs supported [ edit ]

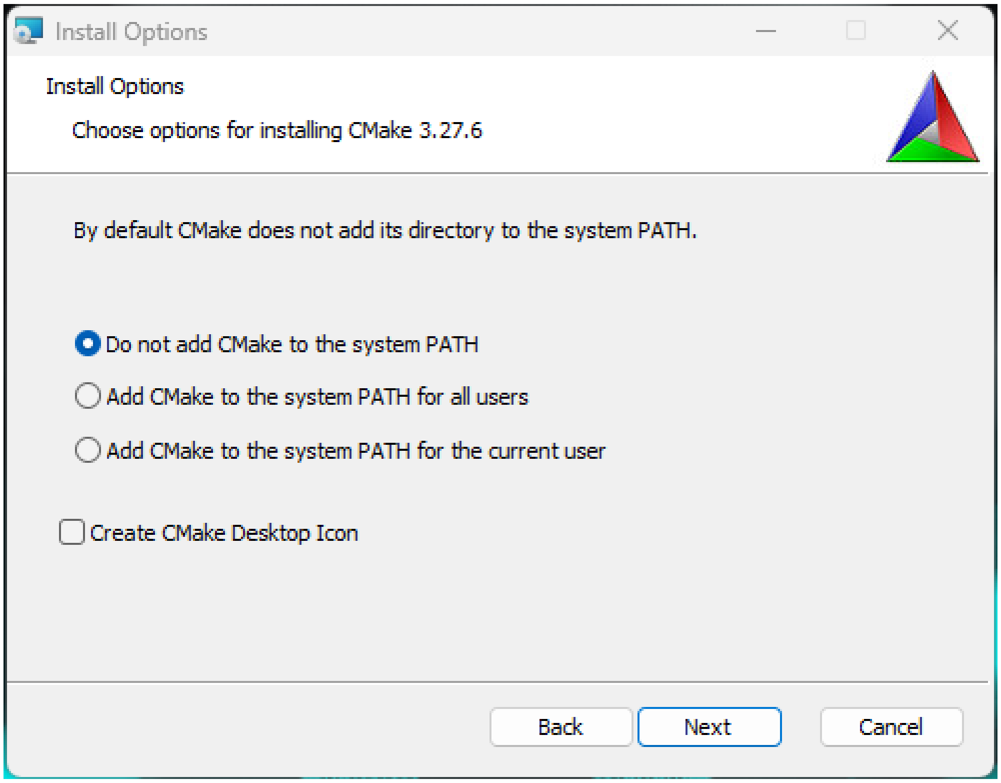
Supported CUDA Compute Capability versions for CUDA SDK version and Microarchitecture (by code name):

Compute Capability (CUDA SDK support vs. Microarchitecture)

| CUDA SDK version(s) | Tesla | Fermi | Kepler (early) | Kepler (late) | Maxwell | Pascal | Volta | Turing | Ampere | Ada Lovelace | Hopper |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.0[34] | 1.0 – 1.1 | | | | | | | | | | |
| 1.1 | 1.0 – 1.1+x | | | | | | | | | | |
| 2.0 | 1.0 – 1.1+x | | | | | | | | | | |
| 2.1 - 2.3.1[35][36][37][38] | 1.0 – 1.3 | | | | | | | | | | |
| 3.0 - 3.1[39][40] | 1.0 | 2.0 | | | | | | | | | |
| 3.2[41] | 1.0 | 2.1 | | | | | | | | | |
| 4.0 - 4.2 | 1.0 | 2.1 | | | | | | | | | |
| 5.0 - 5.5 | 1.0 | 🟩 | 🟩 | 3.5 | | | | | | | |
| 6.0 | 1.0 | 🟩 | 🟩 | 3.5 | | | | | | | |
| 6.5 | 1.1 | 🟩 | 🟩 | 🟩 | 5.x | | | | | | |
| 7.0 - 7.5 | | 2.0 | 🟩 | 🟩 | 5.x | | | | | | |
| 8.0 | | 2.0 | 🟩 | 🟩 | 🟩 | 6.x | | | | | |
| 9.0 - 9.2 | | | 3.0 | 🟩 | 🟩 | 🟩 | 7.0 | | | | |
| 10.0 - 10.2 | | | 3.0 | 🟩 | 🟩 | 🟩 | 🟩 | 7.5 | | | |
| 11.0[42] | | | | 3.5 | 🟩 | 🟩 | 🟩 | 🟩 | 8.0 | | |
| 11.1 - 11.4[43] | | | | 3.5 | 🟩 | 🟩 | 🟩 | 🟩 | 8.6 | | |
| 11.5 - 11.7.1[44] | | | | 3.5 | 🟩 | 🟩 | 🟩 | 🟩 | 8.7 | | |
| 11.8[45] | | | | 3.5 | 🟩 | 🟩 | 🟩 | 🟩 | 🟩 | 🟩 | 9.0 |
| 12.0 - 12.3 | | | | | 5.0 | 🟩 | 🟩 | 🟩 | 🟩 | 🟩 | 9.0 |

# 4 CMake installation:

Install the CMake as usual, and no need to add the path. I am using CMake's latest version, currently 3.27.6



Install Options

Install Options

Choose options for installing CMake 3.27.6

By default CMake does not add its directory to the system PATH.

● Do not add CMake to the system PATH

○ Add CMake to the system PATH for all users

○ Add CMake to the system PATH for the current user

□ Create CMake Desktop Icon

# 5 Visual Studio 2022 installation

You need a working version of Visual Studio. I have tested VS 2017 and 2019 and both worked, but I get some deprecation warnings while compiling some packages, and the warnings can become errors in the future versions of packages, so I updated to VS 2022. I am using the community, free version of VS 2022. Check the installation guide at the link below, and make sure you check ‘Desktop development with C++’.

https://learn.microsoft.com/en-us/cpp/build/vscpp-step-0-installation?view=msvc-170

After VS installation, you need to run it as administrator to build packages, it's easy, follow the link below:

https://www.getfishtank.com/blog/automatically-run-visual-studio-as-administrator

For more information regarding versions, see the link:

https://learn.microsoft.com/en-us/cpp/preprocessor/predefined-macros?view=msvc-160

# 6 Boost installation

You need to download, install, and compile Boost. The most recent version of Boost is 1.83.0. I download this version for VC 2022 (msvc 14.3) from the link below. The file name is: 'boost_1_83_0-msvc-14.3-64.exe'

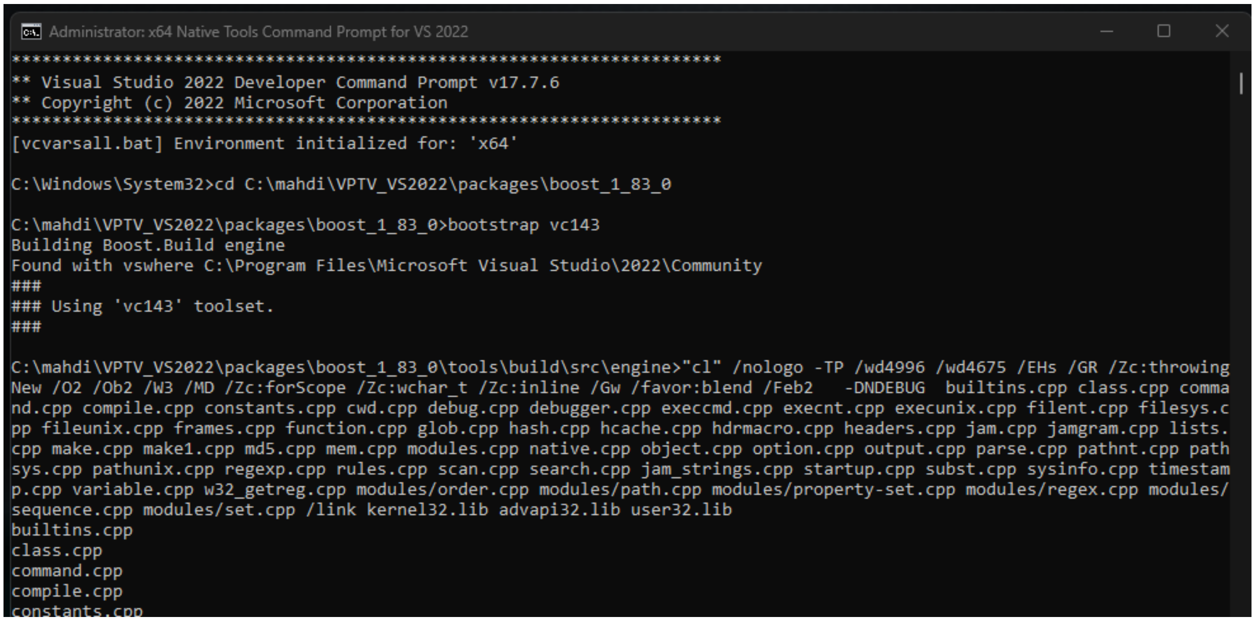
https://sourceforge.net/projects/boost/files/boost-binaries/1.83.0/

You can watch a tutorial at the link below, or follow my installation steps.

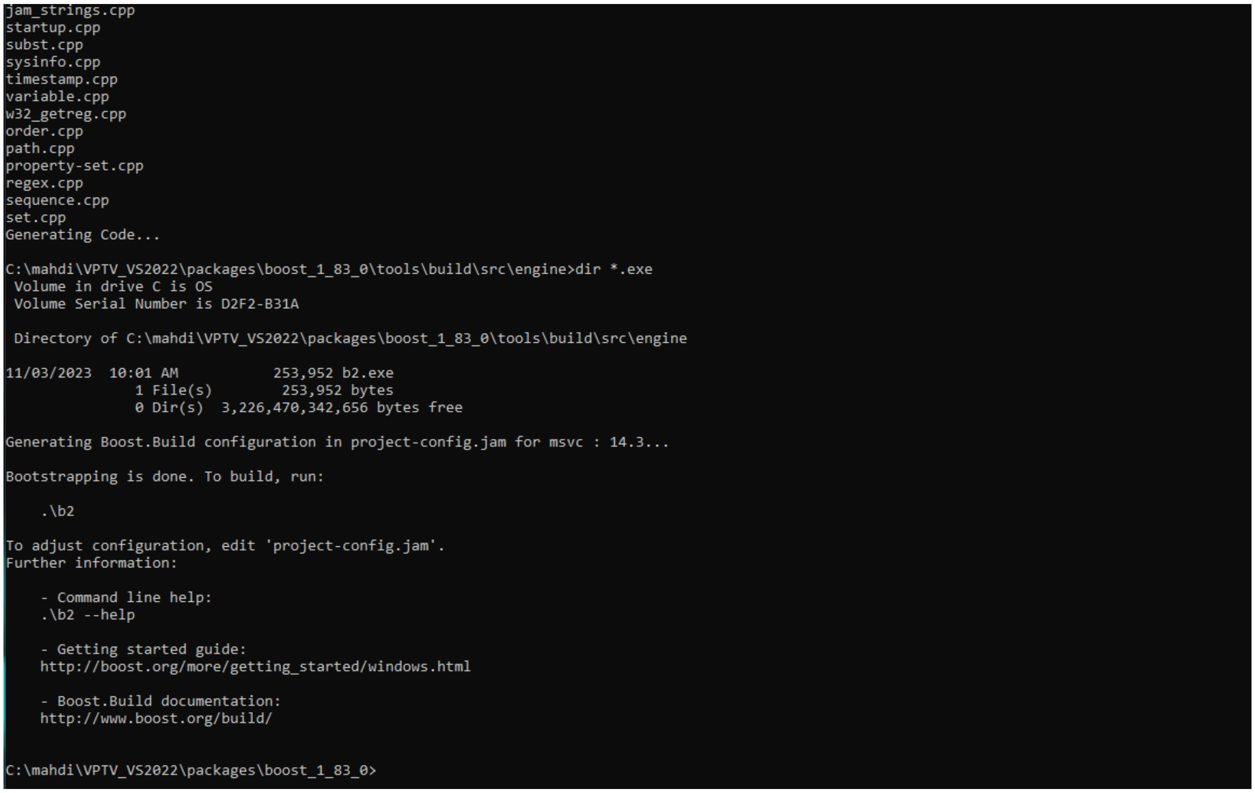
https://www.youtube.com/watch?v=5afpq2TkOHc

Follow the steps below to install Boost and see the figures below:

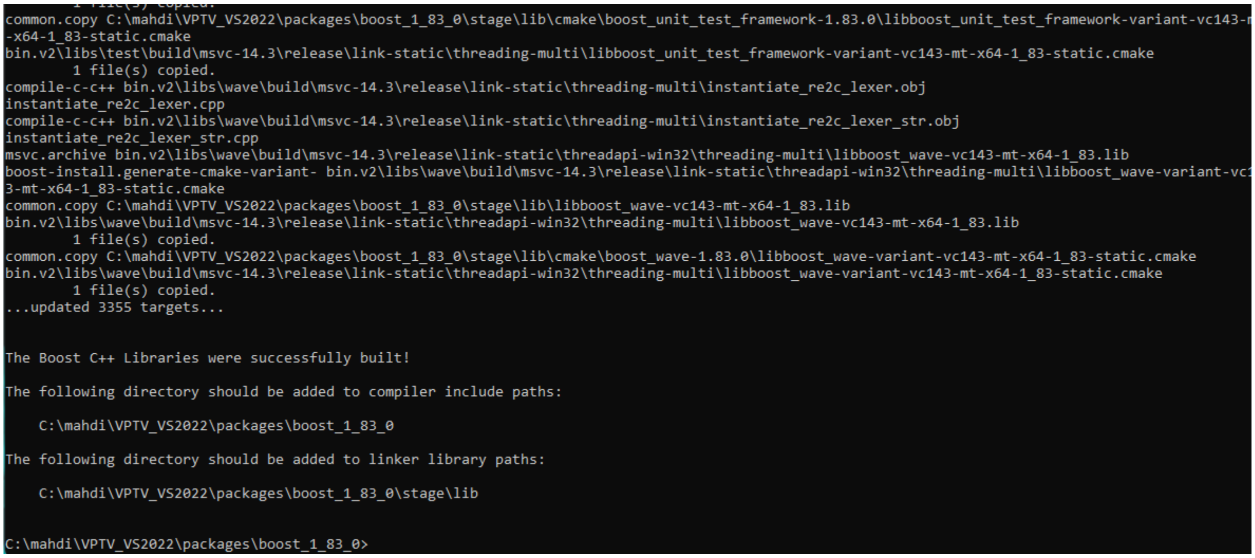
1. Install the package into a directory, mine is at C:\mahdi\VPTV_VS2022\packages\boost_1_83_0
2. From the start, find 'x64 Native Tools Command Prompt for VS 2022' and run it as admin.
3. Type cd and navigate to your boost installation directory.
4. Type 'bootstrap vc143' and enter and wait.
5. Then type '.\b2'. You can also try '.\b2 address-model=64' to compile x64 libraries only.
6. When finished, the boost library is compiled
7. Check and make sure boost static libraries are built, mine is at C:\mahdi\VPTV_VS2022\packages\boost_1_83_0 \stage\lib



```
**********************************************************************
** Visual Studio 2022 Developer Command Prompt v17.7.6
** Copyright (c) 2022 Microsoft Corporation
**********************************************************************
[vcvarsall.bat] Environment initialized for: 'x64'

C:\Windows\System32>cd C:\mahdi\VPTV_VS2022\packages\boost_1_83_0

C:\mahdi\VPTV_VS2022\packages\boost_1_83_0>bootstrap vc143
Building Boost.Build engine
Found with vswhere C:\Program Files\Microsoft Visual Studio\2022\Community
###
### Using 'vc143' toolset.
###

C:\mahdi\VPTV_VS2022\packages\boost_1_83_0\tools\build\src\engine>"cl" /nologo -TP /wd4996 /wd4675 /EHs /GR /Zc:throwing
New /O2 /Ob2 /W3 /MD /Zc:forScope /Zc:wchar_t /Zc:inline /Gw /favor:blend /Feb2   -DNDEBUG  builtins.cpp class.cpp comma
nd.cpp compile.cpp constants.cpp cwd.cpp debug.cpp debugger.cpp execcmd.cpp execnt.cpp execunix.cpp filent.cpp filesys.c
pp fileunix.cpp frames.cpp function.cpp glob.cpp hash.cpp hcache.cpp hdrmacro.cpp headers.cpp jam.cpp jamgram.cpp lists.
cpp make.cpp make1.cpp md5.cpp mem.cpp modules.cpp native.cpp object.cpp option.cpp output.cpp parse.cpp pathnt.cpp path
sys.cpp pathunix.cpp regexp.cpp rules.cpp scan.cpp search.cpp jam_strings.cpp startup.cpp subst.cpp sysinfo.cpp timestam
p.cpp variable.cpp w32_getreg.cpp modules/order.cpp modules/path.cpp modules/property-set.cpp modules/regex.cpp modules/
sequence.cpp modules/set.cpp /link kernel32.lib advapi32.lib user32.lib
builtins.cpp
class.cpp
command.cpp
compile.cpp
constants.cpp
```



```
jam_strings.cpp
startup.cpp
subst.cpp
sysinfo.cpp
timestamp.cpp
variable.cpp
w32_getreg.cpp
order.cpp
path.cpp
property-set.cpp
regex.cpp
sequence.cpp
set.cpp
Generating Code...

C:\mahdi\VPTV_VS2022\packages\boost_1_83_0\tools\build\src\engine>dir *.exe
 Volume in drive C is OS
 Volume Serial Number is D2F2-B31A

 Directory of C:\mahdi\VPTV_VS2022\packages\boost_1_83_0\tools\build\src\engine

11/03/2023  10:01 AM           253,952 b2.exe
               1 File(s)        253,952 bytes
               0 Dir(s)  3,226,470,342,656 bytes free

Generating Boost.Build configuration in project-config.jam for msvc : 14.3...

Bootstrapping is done. To build, run:

    .\b2

To adjust configuration, edit 'project-config.jam'.
Further information:

    - Command line help:
    .\b2 --help

    - Getting started guide:
    http://boost.org/more/getting_started/windows.html

    - Boost.Build documentation:
    http://www.boost.org/build/


C:\mahdi\VPTV_VS2022\packages\boost_1_83_0>
```



```
        1 file(s) copied.
common.copy C:\mahdi\VPTV_VS2022\packages\boost_1_83_0\stage\lib\cmake\boost_unit_test_framework-1.83.0\libboost_unit_test_framework-variant-vc143-m
-x64-1_83-static.cmake
bin.v2\libs\test\build\msvc-14.3\release\link-static\threading-multi\libboost_unit_test_framework-variant-vc143-mt-x64-1_83-static.cmake
        1 file(s) copied.
compile-c-c++ bin.v2\libs\wave\build\msvc-14.3\release\link-static\threading-multi\instantiate_re2c_lexer.obj
instantiate_re2c_lexer.cpp
compile-c-c++ bin.v2\libs\wave\build\msvc-14.3\release\link-static\threading-multi\instantiate_re2c_lexer_str.obj
instantiate_re2c_lexer_str.cpp
msvc.archive bin.v2\libs\wave\build\msvc-14.3\release\link-static\threadapi-win32\threading-multi\libboost_wave-vc143-mt-x64-1_83.lib
boost-install.generate-cmake-variant- bin.v2\libs\wave\build\msvc-14.3\release\link-static\threadapi-win32\threading-multi\libboost_wave-variant-vc1
3-mt-x64-1_83-static.cmake
common.copy C:\mahdi\VPTV_VS2022\packages\boost_1_83_0\stage\lib\libboost_wave-vc143-mt-x64-1_83.lib
bin.v2\libs\wave\build\msvc-14.3\release\link-static\threadapi-win32\threading-multi\libboost_wave-vc143-mt-x64-1_83.lib
        1 file(s) copied.
common.copy C:\mahdi\VPTV_VS2022\packages\boost_1_83_0\stage\lib\cmake\boost_wave-1.83.0\libboost_wave-variant-vc143-mt-x64-1_83-static.cmake
bin.v2\libs\wave\build\msvc-14.3\release\link-static\threadapi-win32\threading-multi\libboost_wave-variant-vc143-mt-x64-1_83-static.cmake
        1 file(s) copied.
...updated 3355 targets...


The Boost C++ Libraries were successfully built!

The following directory should be added to compiler include paths:

    C:\mahdi\VPTV_VS2022\packages\boost_1_83_0

The following directory should be added to linker library paths:

    C:\mahdi\VPTV_VS2022\packages\boost_1_83_0\stage\lib


C:\mahdi\VPTV_VS2022\packages\boost_1_83_0>
```

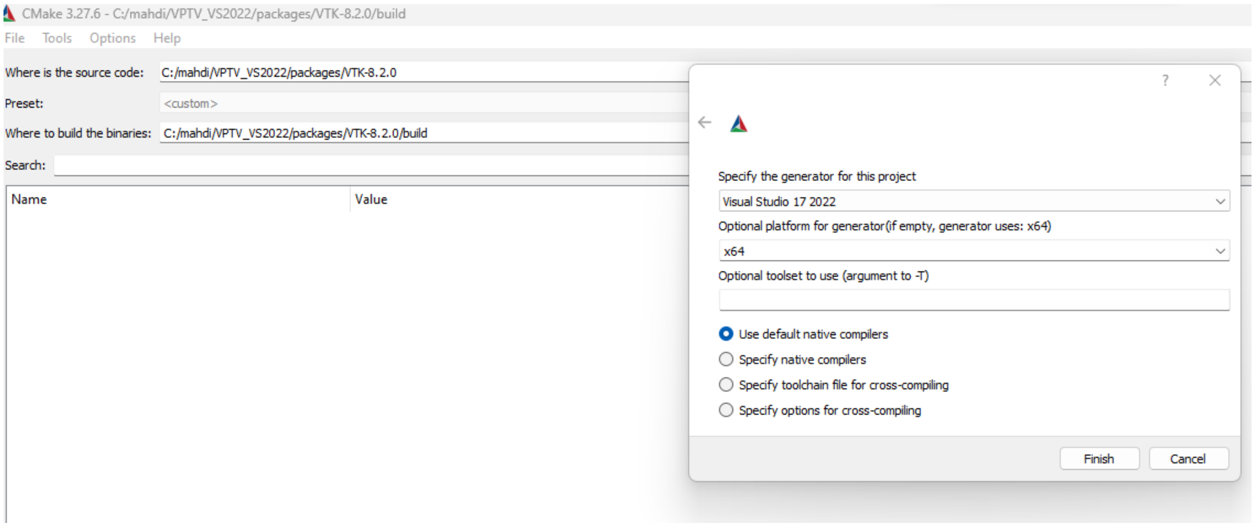
# 7 VTK installation

You need to download and build VTK. The tested version is 8.2.0. Since VTK is only used for virtualization of the software and is working fine, I didn't try updating it. You can watch a tutorial below or follow the instructions. The point is that you should build VTK with QT.

https://www.youtube.com/watch?v=u5-Df1YlxCI

Instructions:

1. Unzip 'VTK-8.2.0.zip' to a directory, and open CMake to that directory:



CMake 3.27.6 - C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0/build

Where is the source code: C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0

Preset: <custom>

Where to build the binaries: C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0/build

Search:

| Name | Value |
|---|---|

Specify the generator for this project: Visual Studio 17 2022

Optional platform for generator(if empty, generator uses: x64): x64

Optional toolset to use (argument to -T):

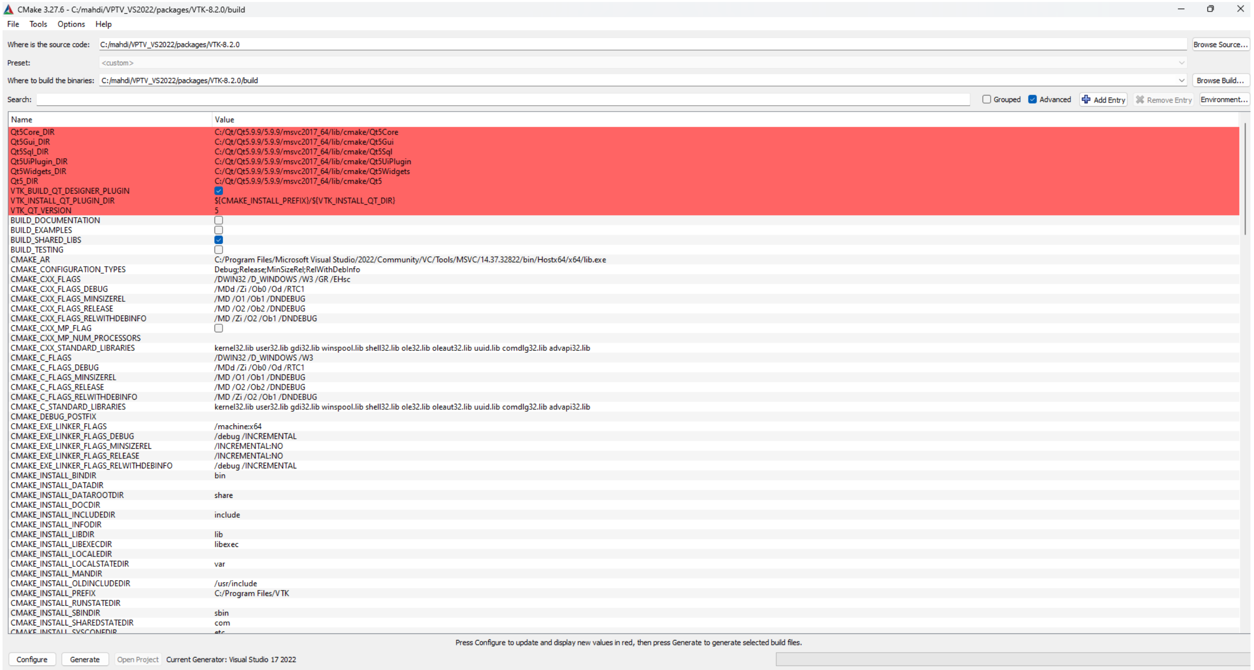
● Use default native compilers

○ Specify native compilers

○ Specify toolchain file for cross-compiling

○ Specify options for cross-compiling

2. Click configure, if you get warnings, that's okay.
3. Find and check vtk_group_qt. Change VTK python version to 3.
4. Click Configure again. Also, note the address for cmake_install_prefix, mine is at: C:/Program Files/VTK
5. Click configure again and make sure Cmake can find the QT directory. If not, add it manually.
6. Click generate, and open the project in VS2022. Make sure VS is in admin mode.
7. In VS, change the configuration type to debug, x64, and right-click on the 'ALL_BUILD' and select build
8. After finishing and seeing the successful build of all modules, right-click on Install and select Build. Make sure you see the build successful message.
9. Go to Windows environment variables and add the following under system variables by clicking on new: variable name: VTK_DIR and variable value: C:\Program Files\VTK
10. Go to Windows environment variables and add the following under 'Path' by selecting path and then edit and then new: %VTK_DIR%\bin



CMake 3.27.6 - C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0/build

Where is the source code: C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0

Preset: <custom>

Where to build the binaries: C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0/build

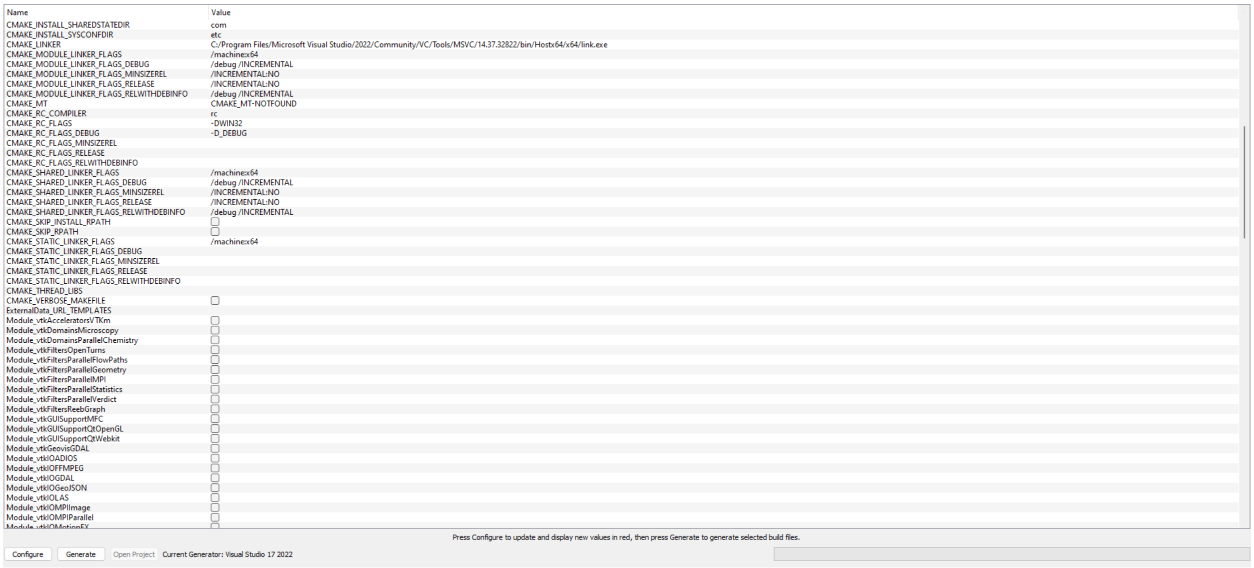
Search:

□ Grouped ☑ Advanced

| Name | Value |
|---|---|
| Qt5Core_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Core |
| Qt5Gui_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Gui |
| Qt5Sql_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Sql |
| Qt5UiPlugin_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5UiPlugin |
| Qt5Widgets_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Widgets |
| Qt5_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5 |
| VTK_BUILD_QT_DESIGNER_PLUGIN | ☑ |
| VTK_INSTALL_QT_PLUGIN_DIR | ${CMAKE_INSTALL_PREFIX}/${VTK_INSTALL_QT_DIR} |
| VTK_QT_VERSION | 5 |
| BUILD_DOCUMENTATION | □ |
| BUILD_EXAMPLES | □ |
| BUILD_SHARED_LIBS | ☑ |
| BUILD_TESTING | □ |
| CMAKE_AR | C:/Program Files/Microsoft Visual Studio/2022/Community/VC/Tools/MSVC/14.37.32822/bin/Hostx64/x64/lib.exe |
| CMAKE_CONFIGURATION_TYPES | Debug;Release;MinSizeRel;RelWithDebInfo |
| CMAKE_CXX_FLAGS | /DWIN32 /D_WINDOWS /W3 /GR /EHsc |
| CMAKE_CXX_FLAGS_DEBUG | /MDd /Zi /Ob0 /Od /RTC1 |
| CMAKE_CXX_FLAGS_MINSIZEREL | /MD /O1 /Ob1 /DNDEBUG |
| CMAKE_CXX_FLAGS_RELEASE | /MD /O2 /Ob2 /DNDEBUG |
| CMAKE_CXX_FLAGS_RELWITHDEBINFO | /MD /Zi /O2 /Ob1 /DNDEBUG |
| CMAKE_CXX_MP_FLAG | □ |
| CMAKE_CXX_MP_NUM_PROCESSORS | |
| CMAKE_CXX_STANDARD_LIBRARIES | kernel32.lib user32.lib gdi32.lib winspool.lib shell32.lib ole32.lib oleaut32.lib uuid.lib comdlg32.lib advapi32.lib |
| CMAKE_C_FLAGS | /DWIN32 /D_WINDOWS /W3 |
| CMAKE_C_FLAGS_DEBUG | /MDd /Zi /Ob0 /Od /RTC1 |
| CMAKE_C_FLAGS_MINSIZEREL | /MD /O1 /Ob1 /DNDEBUG |
| CMAKE_C_FLAGS_RELEASE | /MD /O2 /Ob2 /DNDEBUG |
| CMAKE_C_FLAGS_RELWITHDEBINFO | /MD /Zi /O2 /Ob1 /DNDEBUG |
| CMAKE_C_STANDARD_LIBRARIES | kernel32.lib user32.lib gdi32.lib winspool.lib shell32.lib ole32.lib oleaut32.lib uuid.lib comdlg32.lib advapi32.lib |
| CMAKE_DEBUG_POSTFIX | |
| CMAKE_EXE_LINKER_FLAGS | /machine:x64 |
| CMAKE_EXE_LINKER_FLAGS_DEBUG | /debug /INCREMENTAL |
| CMAKE_EXE_LINKER_FLAGS_MINSIZEREL | /INCREMENTAL:NO |
| CMAKE_EXE_LINKER_FLAGS_RELEASE | /INCREMENTAL:NO |
| CMAKE_EXE_LINKER_FLAGS_RELWITHDEBINFO | /debug /INCREMENTAL |
| CMAKE_INSTALL_BINDIR | bin |
| CMAKE_INSTALL_DATADIR | |
| CMAKE_INSTALL_DATAROOTDIR | share |
| CMAKE_INSTALL_DOCDIR | |
| CMAKE_INSTALL_INCLUDEDIR | include |
| CMAKE_INSTALL_INFODIR | |
| CMAKE_INSTALL_LIBDIR | lib |
| CMAKE_INSTALL_LIBEXECDIR | libexec |
| CMAKE_INSTALL_LOCALEDIR | |
| CMAKE_INSTALL_LOCALSTATEDIR | var |
| CMAKE_INSTALL_MANDIR | |
| CMAKE_INSTALL_OLDINCLUDEDIR | /usr/include |
| CMAKE_INSTALL_PREFIX | C:/Program Files/VTK |
| CMAKE_INSTALL_RUNSTATEDIR | |
| CMAKE_INSTALL_SBINDIR | sbin |
| CMAKE_INSTALL_SHAREDSTATEDIR | com |
| CMAKE_INSTALL_SYSCONFDIR | etc |

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

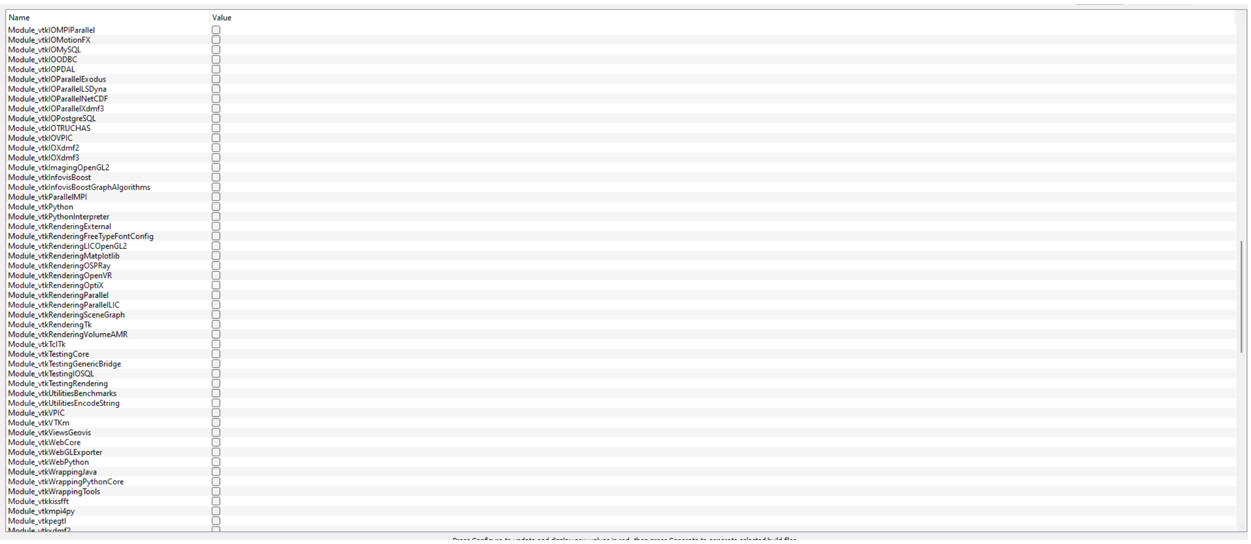
Current Generator: Visual Studio 17 2022



| Name | Value |
|---|---|
| CMAKE_INSTALL_SHAREDSTATEDIR | com |
| CMAKE_INSTALL_SYSCONFDIR | etc |
| CMAKE_LINKER | C:/Program Files/Microsoft Visual Studio/2022/Community/VC/Tools/MSVC/14.37.32822/bin/Hostx64/x64/link.exe |
| CMAKE_MODULE_LINKER_FLAGS | /machine:x64 |
| CMAKE_MODULE_LINKER_FLAGS_DEBUG | /debug /INCREMENTAL |
| CMAKE_MODULE_LINKER_FLAGS_MINSIZEREL | /INCREMENTAL:NO |
| CMAKE_MODULE_LINKER_FLAGS_RELEASE | /INCREMENTAL:NO |
| CMAKE_MODULE_LINKER_FLAGS_RELWITHDEBINFO | /debug /INCREMENTAL |
| CMAKE_MT | CMAKE_MT-NOTFOUND |
| CMAKE_RC_COMPILER | rc |
| CMAKE_RC_FLAGS | -DWIN32 |
| CMAKE_RC_FLAGS_DEBUG | -D_DEBUG |
| CMAKE_RC_FLAGS_MINSIZEREL | |
| CMAKE_RC_FLAGS_RELEASE | |
| CMAKE_RC_FLAGS_RELWITHDEBINFO | |
| CMAKE_SHARED_LINKER_FLAGS | /machine:x64 |
| CMAKE_SHARED_LINKER_FLAGS_DEBUG | /debug /INCREMENTAL |
| CMAKE_SHARED_LINKER_FLAGS_MINSIZEREL | /INCREMENTAL:NO |
| CMAKE_SHARED_LINKER_FLAGS_RELEASE | /INCREMENTAL:NO |
| CMAKE_SHARED_LINKER_FLAGS_RELWITHDEBINFO | /debug /INCREMENTAL |
| CMAKE_SKIP_INSTALL_RPATH | □ |
| CMAKE_SKIP_RPATH | □ |
| CMAKE_STATIC_LINKER_FLAGS | /machine:x64 |
| CMAKE_STATIC_LINKER_FLAGS_DEBUG | |
| CMAKE_STATIC_LINKER_FLAGS_MINSIZEREL | |
| CMAKE_STATIC_LINKER_FLAGS_RELEASE | |
| CMAKE_STATIC_LINKER_FLAGS_RELWITHDEBINFO | |
| CMAKE_THREAD_LIBS | |
| CMAKE_VERBOSE_MAKEFILE | □ |
| ExternalData_URL_TEMPLATES | |
| Module_vtkAcceleratorsVTKm | □ |
| Module_vtkDomainsMicroscopy | □ |
| Module_vtkDomainsParallelChemistry | □ |
| Module_vtkFiltersOpenTurns | □ |
| Module_vtkFiltersParallelFlowPaths | □ |
| Module_vtkFiltersParallelGeometry | □ |
| Module_vtkFiltersParallelMPI | □ |
| Module_vtkFiltersParallelStatistics | □ |
| Module_vtkFiltersParallelVerdict | □ |
| Module_vtkFiltersReebGraph | □ |
| Module_vtkGUISupportMFC | □ |
| Module_vtkGUISupportQtOpenGL | □ |
| Module_vtkGUISupportQtWebkit | □ |
| Module_vtkGeovisGDAL | □ |
| Module_vtkIOADIOS | □ |
| Module_vtkIOFFMPEG | □ |
| Module_vtkIOGDAL | □ |
| Module_vtkIOGeoJSON | □ |
| Module_vtkIOLAS | □ |
| Module_vtkIOMPIImage | □ |
| Module_vtkIOMPIParallel | □ |
| Module_vtkIOMotionFX | □ |

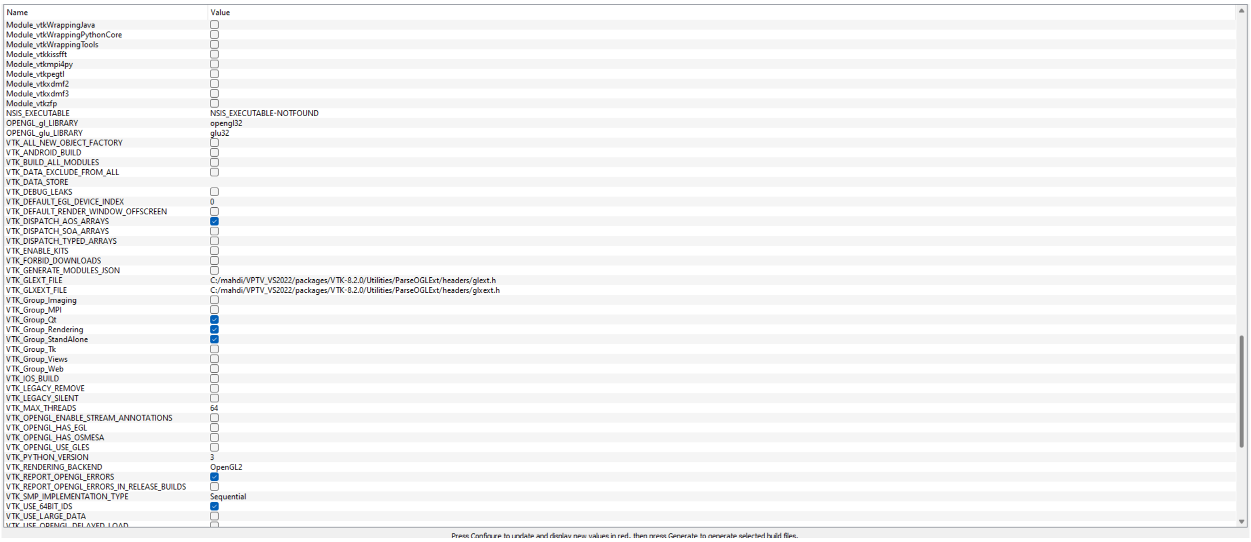
Press Configure to update and display new values in red, then press Generate to generate selected build files.

Current Generator: Visual Studio 17 2022



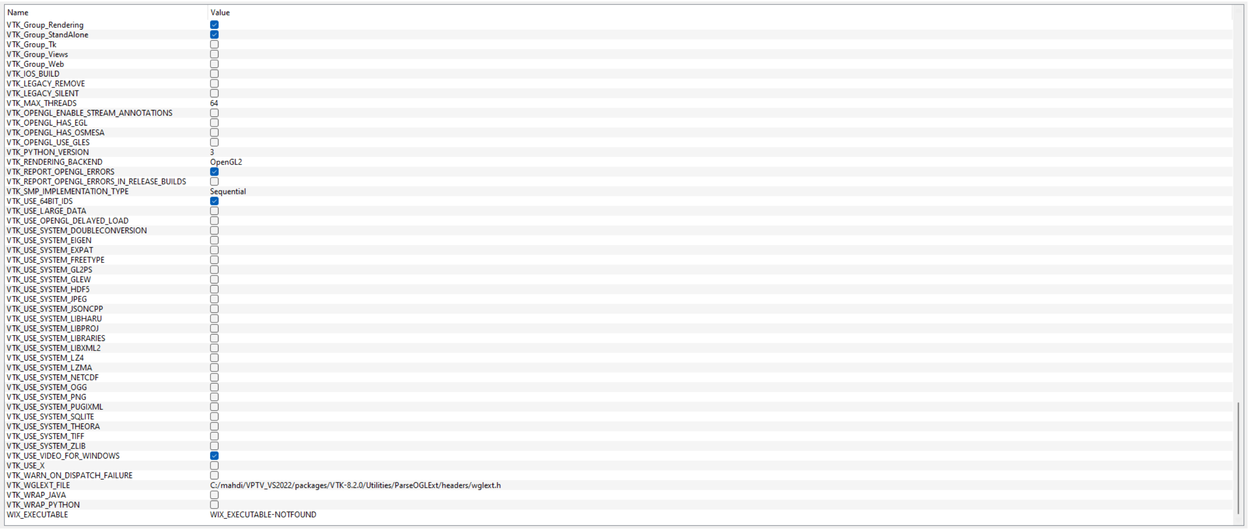
| Name | Value |
|---|---|
| Module_vtkIOMPIParallel | □ |
| Module_vtkIOMotionFX | □ |
| Module_vtkIOMySQL | □ |
| Module_vtkIOODBC | □ |
| Module_vtkIOPDAL | □ |
| Module_vtkIOParallelExodus | □ |
| Module_vtkIOParallelLSDyna | □ |
| Module_vtkIOParallelNetCDF | □ |
| Module_vtkIOParallelXdmf3 | □ |
| Module_vtkIOPostgreSQL | □ |
| Module_vtkIOTRUCHAS | □ |
| Module_vtkIOVPIC | □ |
| Module_vtkIOXdmf2 | □ |
| Module_vtkIOXdmf3 | □ |
| Module_vtkImagingOpenGL2 | □ |
| Module_vtkInfovisBoost | □ |
| Module_vtkInfovisBoostGraphAlgorithms | □ |
| Module_vtkParallelMPI | □ |
| Module_vtkPython | □ |
| Module_vtkPythonInterpreter | □ |
| Module_vtkRenderingExternal | □ |
| Module_vtkRenderingFreeTypeFontConfig | □ |
| Module_vtkRenderingLICOpenGL2 | □ |
| Module_vtkRenderingMatplotlib | □ |
| Module_vtkRenderingOSPRay | □ |
| Module_vtkRenderingOpenVR | □ |
| Module_vtkRenderingOptiX | □ |
| Module_vtkRenderingParallel | □ |
| Module_vtkRenderingParallelLIC | □ |
| Module_vtkRenderingSceneGraph | □ |
| Module_vtkRenderingTk | □ |
| Module_vtkRenderingVolumeAMR | □ |
| Module_vtkTclTk | □ |
| Module_vtkTestingCore | □ |
| Module_vtkTestingGenericBridge | □ |
| Module_vtkTestingIOSQL | □ |
| Module_vtkTestingRendering | □ |
| Module_vtkUtilitiesBenchmarks | □ |
| Module_vtkUtilitiesEncodeString | □ |
| Module_vtkVPIC | □ |
| Module_vtkVTKm | □ |
| Module_vtkViewsGeovis | □ |
| Module_vtkWebCore | □ |
| Module_vtkWebGLExporter | □ |
| Module_vtkWebPython | □ |
| Module_vtkWrappingJava | □ |
| Module_vtkWrappingPythonCore | □ |
| Module_vtkWrappingTools | □ |
| Module_vtkkissfft | □ |
| Module_vtkmpi4py | □ |
| Module_vtkpegtl | □ |
| Module_vtkxdmf2 | □ |

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

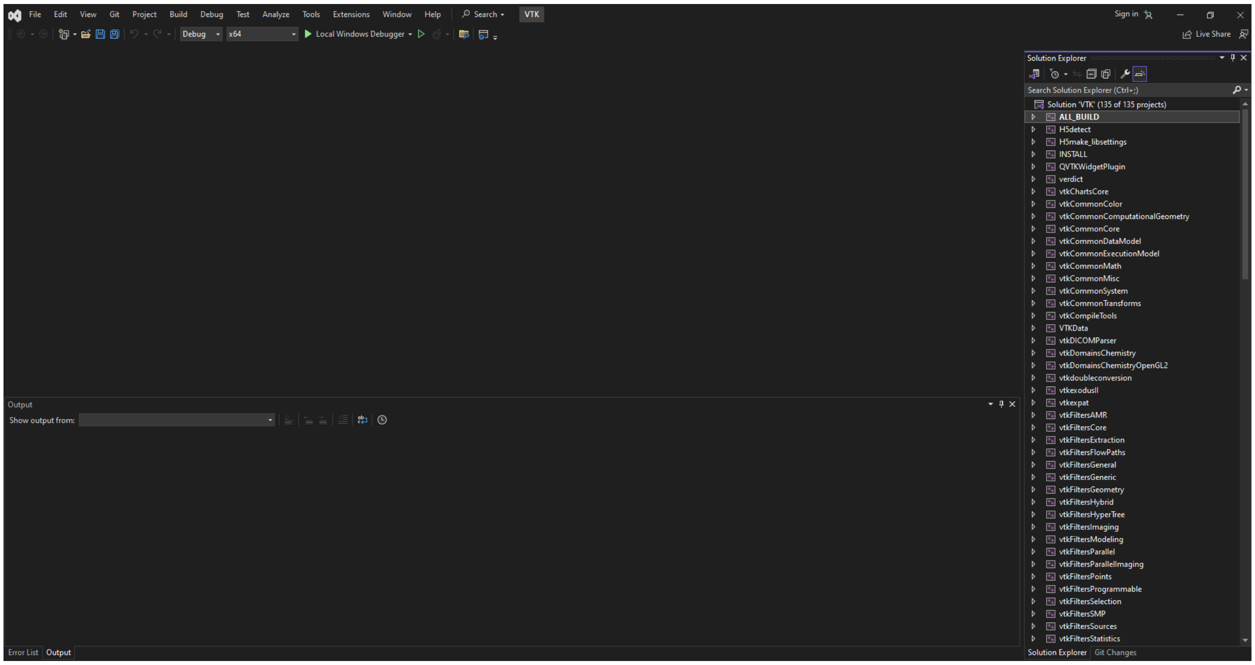


| Name | Value |
|---|---|
| Module_vtkWrappingJava | □ |
| Module_vtkWrappingPythonCore | □ |
| Module_vtkWrappingTools | □ |
| Module_vtkkissfft | □ |
| Module_vtkmpi4py | □ |
| Module_vtkpegtl | □ |
| Module_vtkxdmf2 | □ |
| Module_vtkxdmf3 | □ |
| Module_vtkzfp | □ |
| NSIS_EXECUTABLE | NSIS_EXECUTABLE-NOTFOUND |
| OPENGL_gl_LIBRARY | opengl32 |
| OPENGL_glu_LIBRARY | glu32 |
| VTK_ALL_NEW_OBJECT_FACTORY | □ |
| VTK_ANDROID_BUILD | □ |
| VTK_BUILD_ALL_MODULES | □ |
| VTK_DATA_EXCLUDE_FROM_ALL | □ |
| VTK_DATA_STORE | |
| VTK_DEBUG_LEAKS | □ |
| VTK_DEFAULT_EGL_DEVICE_INDEX | 0 |
| VTK_DEFAULT_RENDER_WINDOW_OFFSCREEN | □ |
| VTK_DISPATCH_AOS_ARRAYS | ☑ |
| VTK_DISPATCH_SOA_ARRAYS | □ |
| VTK_DISPATCH_TYPED_ARRAYS | □ |
| VTK_ENABLE_KITS | □ |
| VTK_FORBID_DOWNLOADS | □ |
| VTK_GENERATE_MODULES_JSON | □ |
| VTK_GLEXT_FILE | C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0/Utilities/ParseOGLExt/headers/glext.h |
| VTK_GLXEXT_FILE | C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0/Utilities/ParseOGLExt/headers/glxext.h |
| VTK_Group_Imaging | □ |
| VTK_Group_MPI | □ |
| VTK_Group_Qt | ☑ |
| VTK_Group_Rendering | ☑ |
| VTK_Group_StandAlone | ☑ |
| VTK_Group_Tk | □ |
| VTK_Group_Views | □ |
| VTK_Group_Web | □ |
| VTK_IOS_BUILD | □ |
| VTK_LEGACY_REMOVE | □ |
| VTK_LEGACY_SILENT | □ |
| VTK_MAX_THREADS | 64 |
| VTK_OPENGL_ENABLE_STREAM_ANNOTATIONS | □ |
| VTK_OPENGL_HAS_EGL | □ |
| VTK_OPENGL_HAS_OSMESA | □ |
| VTK_OPENGL_USE_GLES | □ |
| VTK_PYTHON_VERSION | 3 |
| VTK_RENDERING_BACKEND | OpenGL2 |
| VTK_REPORT_OPENGL_ERRORS | ☑ |
| VTK_REPORT_OPENGL_ERRORS_IN_RELEASE_BUILDS | □ |
| VTK_SMP_IMPLEMENTATION_TYPE | Sequential |
| VTK_USE_64BIT_IDS | ☑ |
| VTK_USE_LARGE_DATA | □ |
| VTK_USE_OPENGL_DELAYED_LOAD | □ |

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



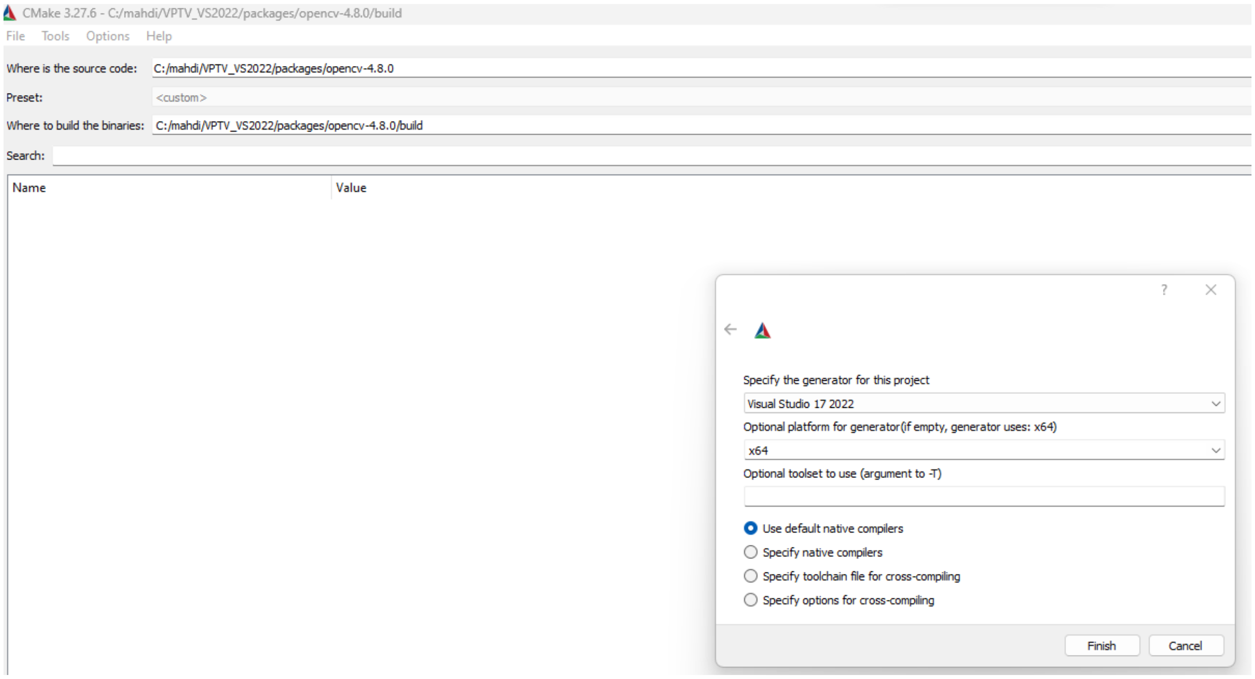
| Name | Value |
|---|---|
| VTK_Group_Rendering | ☑ |
| VTK_Group_StandAlone | ☑ |
| VTK_Group_Tk | □ |
| VTK_Group_Views | □ |
| VTK_Group_Web | □ |
| VTK_IOS_BUILD | □ |
| VTK_LEGACY_REMOVE | □ |
| VTK_LEGACY_SILENT | □ |
| VTK_MAX_THREADS | 64 |
| VTK_OPENGL_ENABLE_STREAM_ANNOTATIONS | □ |
| VTK_OPENGL_HAS_EGL | □ |
| VTK_OPENGL_HAS_OSMESA | □ |
| VTK_OPENGL_USE_GLES | □ |
| VTK_PYTHON_VERSION | 3 |
| VTK_RENDERING_BACKEND | OpenGL2 |
| VTK_REPORT_OPENGL_ERRORS | ☑ |
| VTK_REPORT_OPENGL_ERRORS_IN_RELEASE_BUILDS | □ |
| VTK_SMP_IMPLEMENTATION_TYPE | Sequential |
| VTK_USE_64BIT_IDS | ☑ |
| VTK_USE_LARGE_DATA | □ |
| VTK_USE_OPENGL_DELAYED_LOAD | □ |
| VTK_USE_SYSTEM_DOUBLECONVERSION | □ |
| VTK_USE_SYSTEM_EIGEN | □ |
| VTK_USE_SYSTEM_EXPAT | □ |
| VTK_USE_SYSTEM_FREETYPE | □ |
| VTK_USE_SYSTEM_GL2PS | □ |
| VTK_USE_SYSTEM_GLEW | □ |
| VTK_USE_SYSTEM_HDF5 | □ |
| VTK_USE_SYSTEM_JPEG | □ |
| VTK_USE_SYSTEM_JSONCPP | □ |
| VTK_USE_SYSTEM_LIBHARU | □ |
| VTK_USE_SYSTEM_LIBPROJ | □ |
| VTK_USE_SYSTEM_LIBRARIES | □ |
| VTK_USE_SYSTEM_LIBXML2 | □ |
| VTK_USE_SYSTEM_LZ4 | □ |
| VTK_USE_SYSTEM_LZMA | □ |
| VTK_USE_SYSTEM_NETCDF | □ |
| VTK_USE_SYSTEM_OGG | □ |
| VTK_USE_SYSTEM_PNG | □ |
| VTK_USE_SYSTEM_PUGIXML | □ |
| VTK_USE_SYSTEM_SQLITE | □ |
| VTK_USE_SYSTEM_THEORA | □ |
| VTK_USE_SYSTEM_TIFF | □ |
| VTK_USE_SYSTEM_ZLIB | □ |
| VTK_USE_VIDEO_FOR_WINDOWS | ☑ |
| VTK_USE_X | □ |
| VTK_WARN_ON_DISPATCH_FAILURE | □ |
| VTK_WGLEXT_FILE | C:/mahdi/VPTV_VS2022/packages/VTK-8.2.0/Utilities/ParseOGLExt/headers/wglext.h |
| VTK_WRAP_JAVA | □ |
| VTK_WRAP_PYTHON | □ |
| WIX_EXECUTABLE | WIX_EXECUTABLE-NOTFOUND |


File
Edit
View
Git
Project
Build
Debug
Test
Analyze
Tools
Extensions
Window
Help
Search
VTK
Sign in
Live Share
Debug
x64
Local Windows Debugger
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'VTK' (135 of 135 projects)
ALL_BUILD
H5detect
H5make_libsettings
INSTALL
QVTKWidgetPlugin
verdict
vtkChartsCore
vtkCommonColor
vtkCommonComputationalGeometry
vtkCommonCore
vtkCommonDataModel
vtkCommonExecutionModel
vtkCommonMath
vtkCommonMisc
vtkCommonSystem
vtkCommonTransforms
vtkCompileTools
VTKData
vtkDICOMParser
vtkDomainsChemistry
vtkDomainsChemistryOpenGL2
vtkdoubleconversion
vtkexodusII
vtkexpat
vtkFiltersAMR
vtkFiltersCore
vtkFiltersExtraction
vtkFiltersFlowPaths
vtkFiltersGeneral
vtkFiltersGeneric
vtkFiltersGeometry
vtkFiltersHybrid
vtkFiltersHyperTree
vtkFiltersImaging
vtkFiltersModeling
vtkFiltersParallel
vtkFiltersParallelImaging
vtkFiltersPoints
vtkFiltersProgrammable
vtkFiltersSelection
vtkFiltersSMP
vtkFiltersSources
vtkFiltersStatistics
Output
Show output from:
Error List
Output
Solution Explorer
Git Changes

# 8 OpenCV installation:

Although the GitHub page mentions OpenCV 4.1.1, I am using opencv latest version, 4.8.0. you need to compile opencv on your PC with the required settings:

1. download and extract opencv 4.8.0 and opencv-contrib 4.8.0 zip files to some directory
2. open CMake and navigate to the extracted directory of opencv, and for the build directory, enter as shown in the image.
3. click on configure and finish, as shown below. You may get some warnings, but that's okay. If you get errors, that's problematic.



CMake 3.27.6 - C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build

Where is the source code: C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0

Preset: <custom>

Where to build the binaries: C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build

Search:

Specify the generator for this project: Visual Studio 17 2022

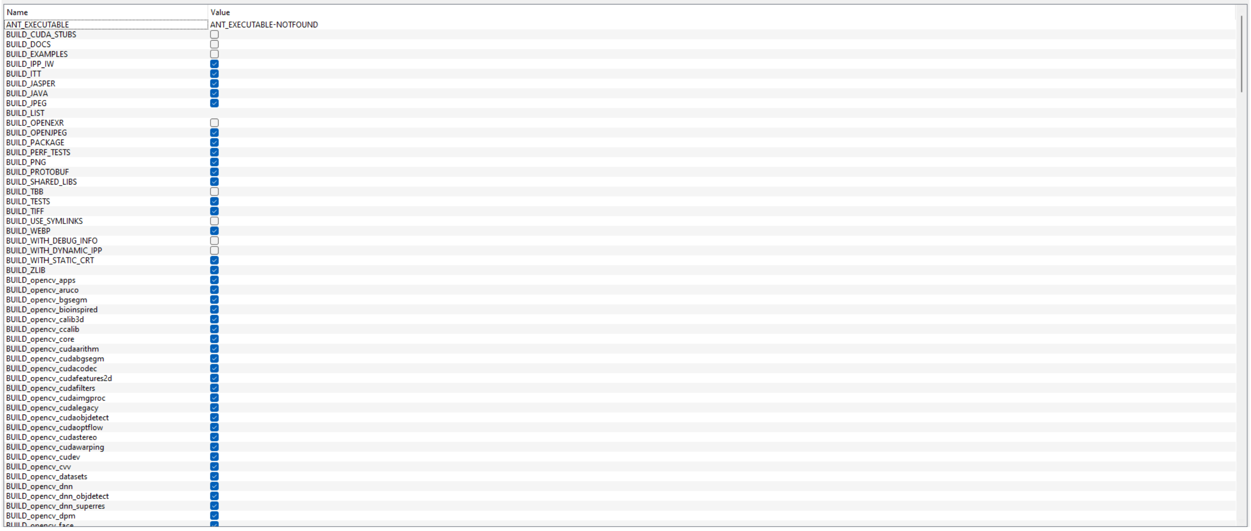
Optional platform for generator(if empty, generator uses: x64): x64

Optional toolset to use (argument to -T):

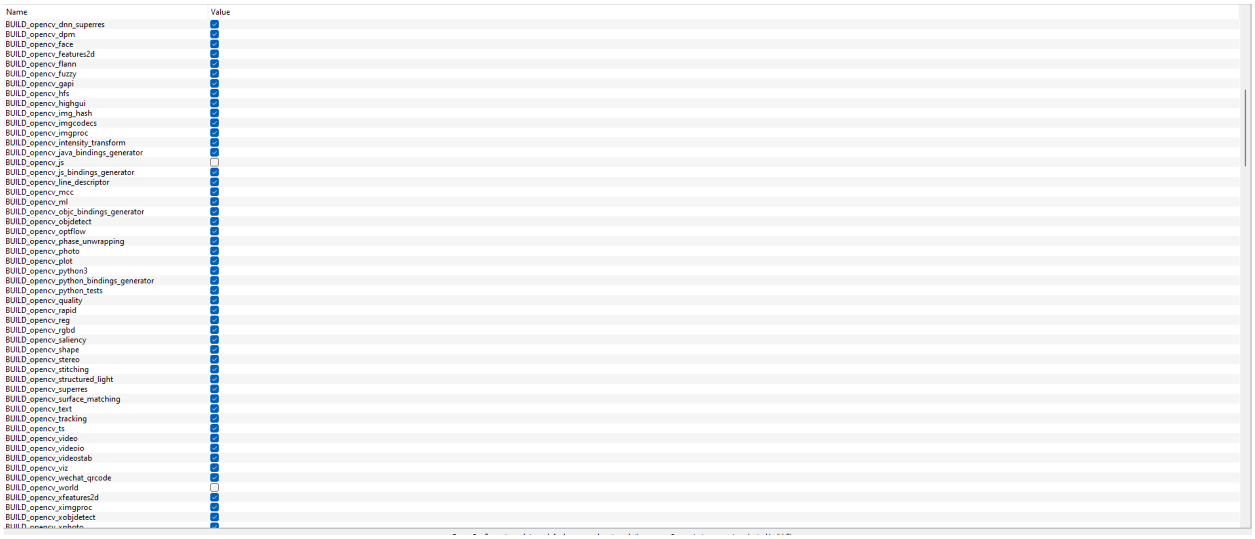
● Use default native compilers
○ Specify native compilers
○ Specify toolchain file for cross-compiling
○ Specify options for cross-compiling

4. in the search box, find and set the values of keys as below:
5. check enable fast math
6. check opencv dnn cuda
7. check with cuda
8. opencv extra modules path should point to the contrib folder/modules. Use the 3 dots to browse by cmake.
9. I have python 3.12, so check opencv python3 version
10. if you don't have numpy installed on your python, first install and then 'python 3 include dirs' should point to numpy folder. Mine is at: C:/Python/Python312/Lib/site-packages/numpy/core/include
11. change cmake configuration types to release only
12. check with QT, and make sure CMake shows your QT directory
13. check with VTK, and make sure CMake shows your VTK directory
14. hit configure again
15. change cuda arch bin to your graphics card suitable version in the link. I have Nvidia 4090, so I keep arch bin 8.9. link: https://en.wikipedia.org/wiki/CUDA
16. Check Cuda fast math

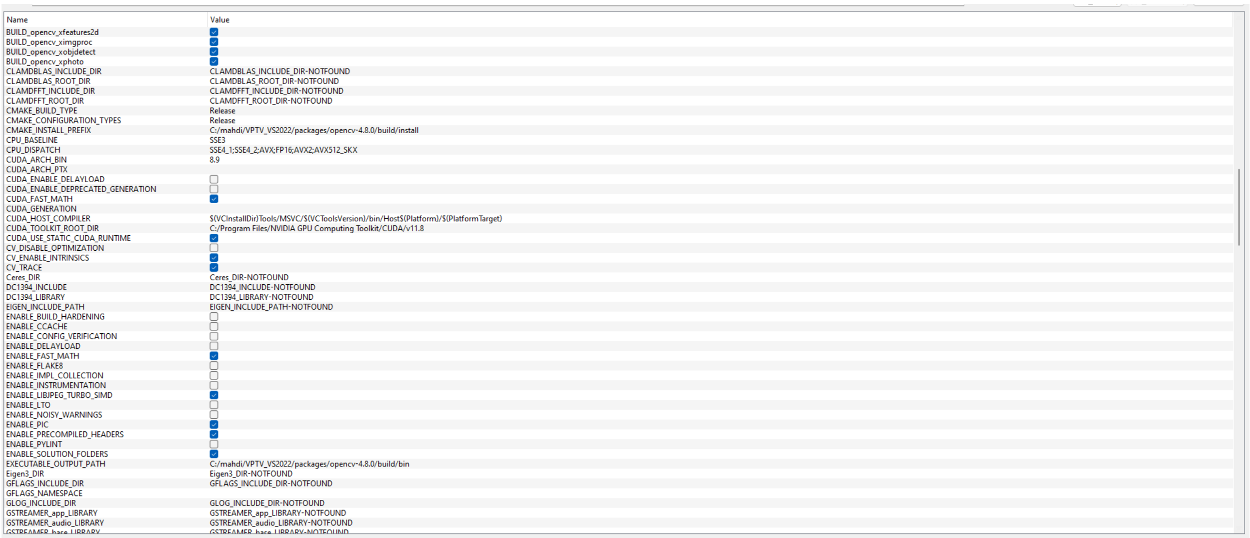
17. hit configure.
18. click generate and then open project to open the file ‘OpenCV.sln’ in the build directory with VS2022
19. if you have any problems, refer to screenshots.
20. In VS2022, find cmaketargets in the tree, right click on ALL BUILD and select build
21. Wait until finish and successful build of all modules, and then right click on install and select build
22. If you need it, you can find my CMake log in the file ‘ocv 4.7.8 log.txt’ attached to this document.



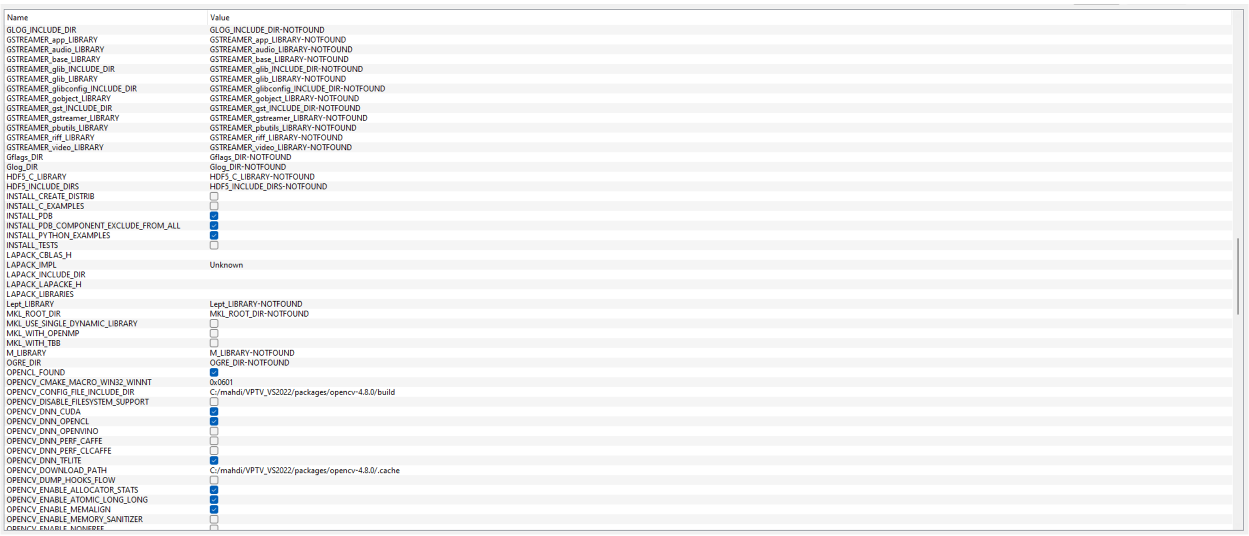
| Name | Value |
|---|---|
| ANT_EXECUTABLE | ANT_EXECUTABLE-NOTFOUND |
| BUILD_CUDA_STUBS | ☐ |
| BUILD_DOCS | ☐ |
| BUILD_EXAMPLES | ☐ |
| BUILD_IPP_IW | ☑ |
| BUILD_ITT | ☑ |
| BUILD_JASPER | ☑ |
| BUILD_JAVA | ☑ |
| BUILD_JPEG | ☑ |
| BUILD_LIST | |
| BUILD_OPENEXR | ☐ |
| BUILD_OPENJPEG | ☑ |
| BUILD_PACKAGE | ☑ |
| BUILD_PERF_TESTS | ☑ |
| BUILD_PNG | ☑ |
| BUILD_PROTOBUF | ☑ |
| BUILD_SHARED_LIBS | ☑ |
| BUILD_TBB | ☐ |
| BUILD_TESTS | ☑ |
| BUILD_TIFF | ☑ |
| BUILD_USE_SYMLINKS | ☐ |
| BUILD_WEBP | ☑ |
| BUILD_WITH_DEBUG_INFO | ☐ |
| BUILD_WITH_DYNAMIC_IPP | ☐ |
| BUILD_WITH_STATIC_CRT | ☑ |
| BUILD_ZLIB | ☑ |
| BUILD_opencv_apps | ☑ |
| BUILD_opencv_aruco | ☑ |
| BUILD_opencv_bgsegm | ☑ |
| BUILD_opencv_bioinspired | ☑ |
| BUILD_opencv_calib3d | ☑ |
| BUILD_opencv_ccalib | ☑ |
| BUILD_opencv_core | ☑ |
| BUILD_opencv_cudaarithm | ☑ |
| BUILD_opencv_cudabgsegm | ☑ |
| BUILD_opencv_cudacodec | ☑ |
| BUILD_opencv_cudafeatures2d | ☑ |
| BUILD_opencv_cudafilters | ☑ |
| BUILD_opencv_cudaimgproc | ☑ |
| BUILD_opencv_cudalegacy | ☑ |
| BUILD_opencv_cudaobjdetect | ☑ |
| BUILD_opencv_cudaoptflow | ☑ |
| BUILD_opencv_cudastereo | ☑ |
| BUILD_opencv_cudawarping | ☑ |
| BUILD_opencv_cudev | ☑ |
| BUILD_opencv_cvv | ☑ |
| BUILD_opencv_datasets | ☑ |
| BUILD_opencv_dnn | ☑ |
| BUILD_opencv_dnn_objdetect | ☑ |
| BUILD_opencv_dnn_superres | ☑ |
| BUILD_opencv_dpm | ☑ |
| BUILD_opencv_face | ☑ |



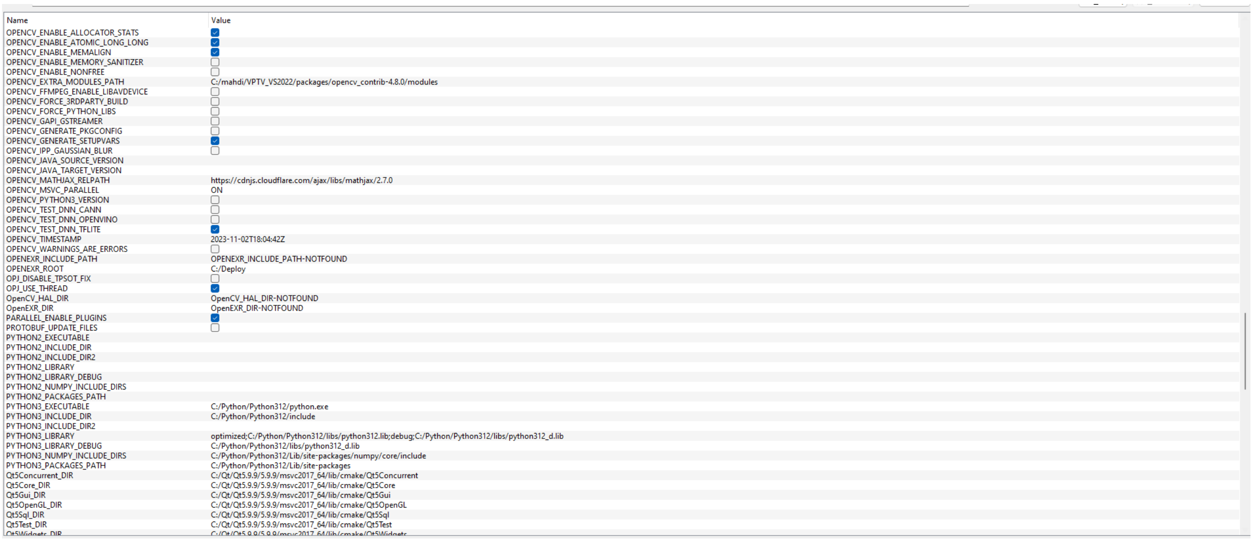
| Name | Value |
|---|---|
| BUILD_opencv_dnn_superres | ☑ |
| BUILD_opencv_dpm | ☑ |
| BUILD_opencv_face | ☑ |
| BUILD_opencv_features2d | ☑ |
| BUILD_opencv_flann | ☑ |
| BUILD_opencv_fuzzy | ☑ |
| BUILD_opencv_gapi | ☑ |
| BUILD_opencv_hfs | ☑ |
| BUILD_opencv_highgui | ☑ |
| BUILD_opencv_img_hash | ☑ |
| BUILD_opencv_imgcodecs | ☑ |
| BUILD_opencv_imgproc | ☑ |
| BUILD_opencv_intensity_transform | ☑ |
| BUILD_opencv_java_bindings_generator | ☑ |
| BUILD_opencv_js | ☐ |
| BUILD_opencv_js_bindings_generator | ☑ |
| BUILD_opencv_line_descriptor | ☑ |
| BUILD_opencv_mcc | ☑ |
| BUILD_opencv_ml | ☑ |
| BUILD_opencv_objc_bindings_generator | ☑ |
| BUILD_opencv_objdetect | ☑ |
| BUILD_opencv_optflow | ☑ |
| BUILD_opencv_phase_unwrapping | ☑ |
| BUILD_opencv_photo | ☑ |
| BUILD_opencv_plot | ☑ |
| BUILD_opencv_python3 | ☑ |
| BUILD_opencv_python_bindings_generator | ☑ |
| BUILD_opencv_python_tests | ☑ |
| BUILD_opencv_quality | ☑ |
| BUILD_opencv_rapid | ☑ |
| BUILD_opencv_reg | ☑ |
| BUILD_opencv_rgbd | ☑ |
| BUILD_opencv_saliency | ☑ |
| BUILD_opencv_shape | ☑ |
| BUILD_opencv_stereo | ☑ |
| BUILD_opencv_stitching | ☑ |
| BUILD_opencv_structured_light | ☑ |
| BUILD_opencv_superres | ☑ |
| BUILD_opencv_surface_matching | ☑ |
| BUILD_opencv_text | ☑ |
| BUILD_opencv_tracking | ☑ |
| BUILD_opencv_ts | ☑ |
| BUILD_opencv_video | ☑ |
| BUILD_opencv_videoio | ☑ |
| BUILD_opencv_videostab | ☑ |
| BUILD_opencv_viz | ☑ |
| BUILD_opencv_wechat_qrcode | ☑ |
| BUILD_opencv_world | ☐ |
| BUILD_opencv_xfeatures2d | ☑ |
| BUILD_opencv_ximgproc | ☑ |
| BUILD_opencv_xobjdetect | ☑ |
| BUILD_opencv_xphoto | ☑ |



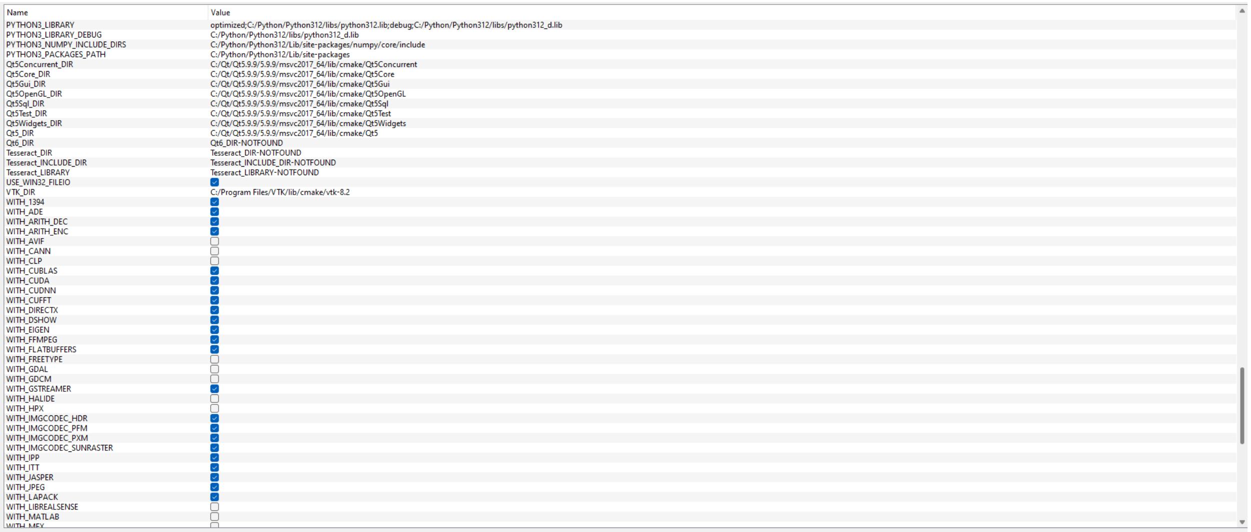
| Name | Value |
|---|---|
| BUILD_opencv_xfeatures2d | ☑ |
| BUILD_opencv_ximgproc | ☑ |
| BUILD_opencv_xobjdetect | ☑ |
| BUILD_opencv_xphoto | ☑ |
| CLAMDBLAS_INCLUDE_DIR | CLAMDBLAS_INCLUDE_DIR-NOTFOUND |
| CLAMDBLAS_ROOT_DIR | CLAMDBLAS_ROOT_DIR-NOTFOUND |
| CLAMDFFT_INCLUDE_DIR | CLAMDFFT_INCLUDE_DIR-NOTFOUND |
| CLAMDFFT_ROOT_DIR | CLAMDFFT_ROOT_DIR-NOTFOUND |
| CMAKE_BUILD_TYPE | Release |
| CMAKE_CONFIGURATION_TYPES | Release |
| CMAKE_INSTALL_PREFIX | C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build/install |
| CPU_BASELINE | SSE3 |
| CPU_DISPATCH | SSE4_1;SSE4_2;AVX;FP16;AVX2;AVX512_SKX |
| CUDA_ARCH_BIN | 8.9 |
| CUDA_ARCH_PTX | |
| CUDA_ENABLE_DELAYLOAD | □ |
| CUDA_ENABLE_DEPRECATED_GENERATION | □ |
| CUDA_FAST_MATH | ☑ |
| CUDA_GENERATION | |
| CUDA_HOST_COMPILER | $(VCInstallDir)Tools/MSVC/$(VCToolsVersion)/bin/Host$(Platform)/$(PlatformTarget) |
| CUDA_TOOLKIT_ROOT_DIR | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8 |
| CUDA_USE_STATIC_CUDA_RUNTIME | ☑ |
| CV_DISABLE_OPTIMIZATION | □ |
| CV_ENABLE_INTRINSICS | ☑ |
| CV_TRACE | ☑ |
| Ceres_DIR | Ceres_DIR-NOTFOUND |
| DC1394_INCLUDE | DC1394_INCLUDE-NOTFOUND |
| DC1394_LIBRARY | DC1394_LIBRARY-NOTFOUND |
| EIGEN_INCLUDE_PATH | EIGEN_INCLUDE_PATH-NOTFOUND |
| ENABLE_BUILD_HARDENING | □ |
| ENABLE_CCACHE | □ |
| ENABLE_CONFIG_VERIFICATION | □ |
| ENABLE_DELAYLOAD | □ |
| ENABLE_FAST_MATH | ☑ |
| ENABLE_FLAKE8 | □ |
| ENABLE_IMPL_COLLECTION | □ |
| ENABLE_INSTRUMENTATION | □ |
| ENABLE_LIBJPEG_TURBO_SIMD | ☑ |
| ENABLE_LTO | □ |
| ENABLE_NOISY_WARNINGS | □ |
| ENABLE_PIC | ☑ |
| ENABLE_PRECOMPILED_HEADERS | ☑ |
| ENABLE_PYLINT | □ |
| ENABLE_SOLUTION_FOLDERS | ☑ |
| EXECUTABLE_OUTPUT_PATH | C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build/bin |
| Eigen3_DIR | Eigen3_DIR-NOTFOUND |
| GFLAGS_INCLUDE_DIR | GFLAGS_INCLUDE_DIR-NOTFOUND |
| GFLAGS_NAMESPACE | |
| GLOG_INCLUDE_DIR | GLOG_INCLUDE_DIR-NOTFOUND |
| GSTREAMER_app_LIBRARY | GSTREAMER_app_LIBRARY-NOTFOUND |
| GSTREAMER_audio_LIBRARY | GSTREAMER_audio_LIBRARY-NOTFOUND |
| GSTREAMER_base_LIBRARY | GSTREAMER_base_LIBRARY-NOTFOUND |



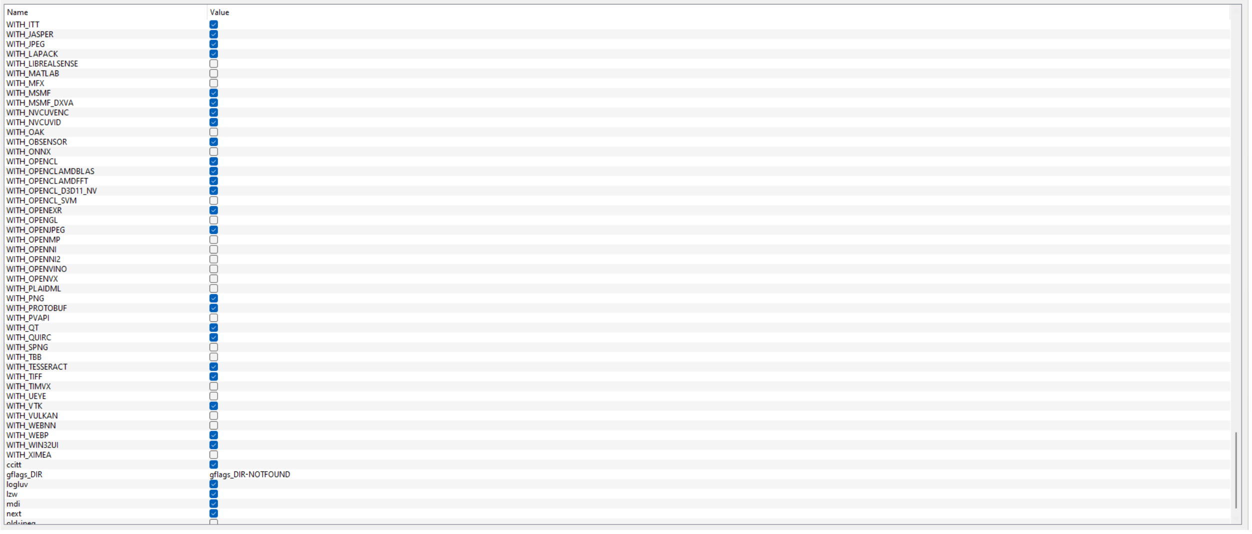
| Name | Value |
|---|---|
| GLOG_INCLUDE_DIR | GLOG_INCLUDE_DIR-NOTFOUND |
| GSTREAMER_app_LIBRARY | GSTREAMER_app_LIBRARY-NOTFOUND |
| GSTREAMER_audio_LIBRARY | GSTREAMER_audio_LIBRARY-NOTFOUND |
| GSTREAMER_base_LIBRARY | GSTREAMER_base_LIBRARY-NOTFOUND |
| GSTREAMER_glib_INCLUDE_DIR | GSTREAMER_glib_INCLUDE_DIR-NOTFOUND |
| GSTREAMER_glib_LIBRARY | GSTREAMER_glib_LIBRARY-NOTFOUND |
| GSTREAMER_glibconfig_INCLUDE_DIR | GSTREAMER_glibconfig_INCLUDE_DIR-NOTFOUND |
| GSTREAMER_gobject_LIBRARY | GSTREAMER_gobject_LIBRARY-NOTFOUND |
| GSTREAMER_gst_INCLUDE_DIR | GSTREAMER_gst_INCLUDE_DIR-NOTFOUND |
| GSTREAMER_gstreamer_LIBRARY | GSTREAMER_gstreamer_LIBRARY-NOTFOUND |
| GSTREAMER_pbutils_LIBRARY | GSTREAMER_pbutils_LIBRARY-NOTFOUND |
| GSTREAMER_riff_LIBRARY | GSTREAMER_riff_LIBRARY-NOTFOUND |
| GSTREAMER_video_LIBRARY | GSTREAMER_video_LIBRARY-NOTFOUND |
| Gflags_DIR | Gflags_DIR-NOTFOUND |
| Glog_DIR | Glog_DIR-NOTFOUND |
| HDF5_C_LIBRARY | HDF5_C_LIBRARY-NOTFOUND |
| HDF5_INCLUDE_DIRS | HDF5_INCLUDE_DIRS-NOTFOUND |
| INSTALL_CREATE_DISTRIB | □ |
| INSTALL_C_EXAMPLES | □ |
| INSTALL_PDB | ☑ |
| INSTALL_PDB_COMPONENT_EXCLUDE_FROM_ALL | ☑ |
| INSTALL_PYTHON_EXAMPLES | ☑ |
| INSTALL_TESTS | □ |
| LAPACK_CBLAS_H | |
| LAPACK_IMPL | Unknown |
| LAPACK_INCLUDE_DIR | |
| LAPACK_LAPACKE_H | |
| LAPACK_LIBRARIES | |
| Lept_LIBRARY | Lept_LIBRARY-NOTFOUND |
| MKL_ROOT_DIR | MKL_ROOT_DIR-NOTFOUND |
| MKL_USE_SINGLE_DYNAMIC_LIBRARY | □ |
| MKL_WITH_OPENMP | □ |
| MKL_WITH_TBB | □ |
| M_LIBRARY | M_LIBRARY-NOTFOUND |
| OGRE_DIR | OGRE_DIR-NOTFOUND |
| OPENCL_FOUND | ☑ |
| OPENCV_CMAKE_MACRO_WIN32_WINNT | 0x0601 |
| OPENCV_CONFIG_FILE_INCLUDE_DIR | C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build |
| OPENCV_DISABLE_FILESYSTEM_SUPPORT | □ |
| OPENCV_DNN_CUDA | ☑ |
| OPENCV_DNN_OPENCL | ☑ |
| OPENCV_DNN_OPENVINO | □ |
| OPENCV_DNN_PERF_CAFFE | □ |
| OPENCV_DNN_PERF_CLCAFFE | □ |
| OPENCV_DNN_TFLITE | ☑ |
| OPENCV_DOWNLOAD_PATH | C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/.cache |
| OPENCV_DUMP_HOOKS_FLOW | □ |
| OPENCV_ENABLE_ALLOCATOR_STATS | ☑ |
| OPENCV_ENABLE_ATOMIC_LONG_LONG | ☑ |
| OPENCV_ENABLE_MEMALIGN | ☑ |
| OPENCV_ENABLE_MEMORY_SANITIZER | □ |
| OPENCV_ENABLE_NONFREE | □ |



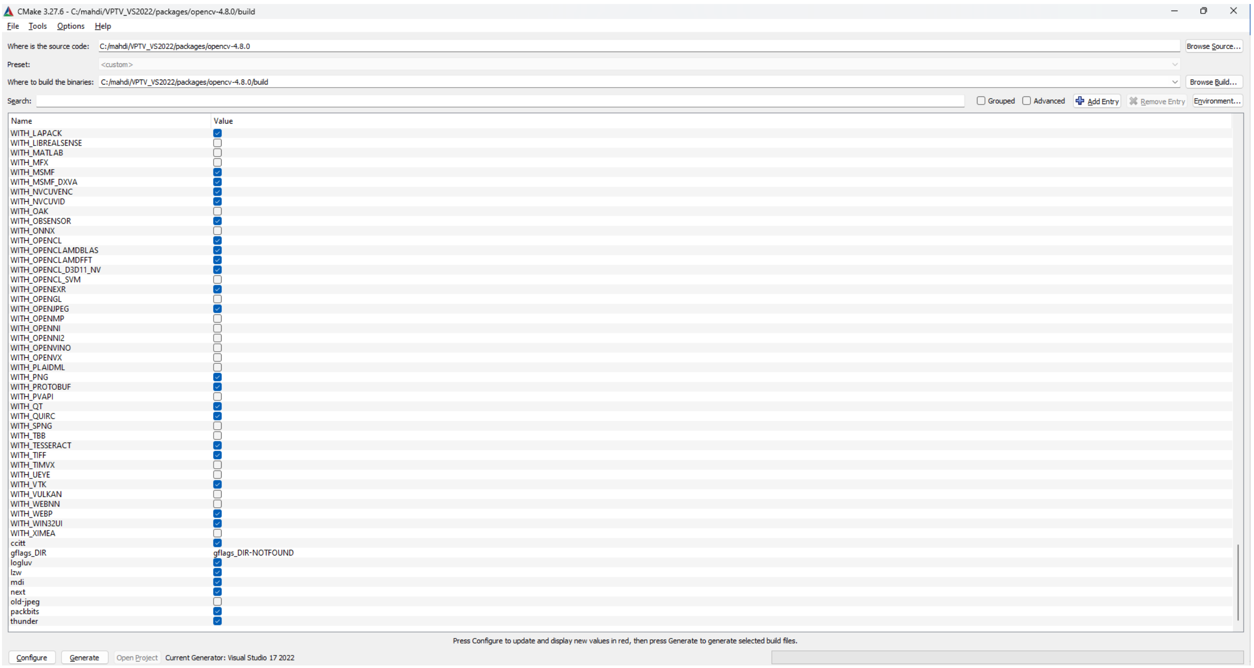
| Name | Value |
|---|---|
| OPENCV_ENABLE_ALLOCATOR_STATS | ☑ |
| OPENCV_ENABLE_ATOMIC_LONG_LONG | ☑ |
| OPENCV_ENABLE_MEMALIGN | ☑ |
| OPENCV_ENABLE_MEMORY_SANITIZER | □ |
| OPENCV_ENABLE_NONFREE | □ |
| OPENCV_EXTRA_MODULES_PATH | C:/mahdi/VPTV_VS2022/packages/opencv_contrib-4.8.0/modules |
| OPENCV_FFMPEG_ENABLE_LIBAVDEVICE | □ |
| OPENCV_FORCE_3RDPARTY_BUILD | □ |
| OPENCV_FORCE_PYTHON_LIBS | □ |
| OPENCV_GAPI_GSTREAMER | □ |
| OPENCV_GENERATE_PKGCONFIG | □ |
| OPENCV_GENERATE_SETUPVARS | ☑ |
| OPENCV_IPP_GAUSSIAN_BLUR | □ |
| OPENCV_JAVA_SOURCE_VERSION | |
| OPENCV_JAVA_TARGET_VERSION | |
| OPENCV_MATHJAX_RELPATH | https://cdnjs.cloudflare.com/ajax/libs/mathjax/2.7.0 |
| OPENCV_MSVC_PARALLEL | ON |
| OPENCV_PYTHON3_VERSION | □ |
| OPENCV_TEST_DNN_CANN | □ |
| OPENCV_TEST_DNN_OPENVINO | □ |
| OPENCV_TEST_DNN_TFLITE | ☑ |
| OPENCV_TIMESTAMP | 2023-11-02T18:04:42Z |
| OPENCV_WARNINGS_ARE_ERRORS | □ |
| OPENEXR_INCLUDE_PATH | OPENEXR_INCLUDE_PATH-NOTFOUND |
| OPENEXR_ROOT | C:/Deploy |
| OPJ_DISABLE_TPSOT_FIX | □ |
| OPJ_USE_THREAD | ☑ |
| OpenCV_HAL_DIR | OpenCV_HAL_DIR-NOTFOUND |
| OpenEXR_DIR | OpenEXR_DIR-NOTFOUND |
| PARALLEL_ENABLE_PLUGINS | ☑ |
| PROTOBUF_UPDATE_FILES | □ |
| PYTHON2_EXECUTABLE | |
| PYTHON2_INCLUDE_DIR | |
| PYTHON2_INCLUDE_DIR2 | |
| PYTHON2_LIBRARY | |
| PYTHON2_LIBRARY_DEBUG | |
| PYTHON2_NUMPY_INCLUDE_DIRS | |
| PYTHON2_PACKAGES_PATH | |
| PYTHON3_EXECUTABLE | C:/Python/Python312/python.exe |
| PYTHON3_INCLUDE_DIR | C:/Python/Python312/include |
| PYTHON3_INCLUDE_DIR2 | |
| PYTHON3_LIBRARY | optimized;C:/Python/Python312/libs/python312.lib;debug;C:/Python/Python312/libs/python312_d.lib |
| PYTHON3_LIBRARY_DEBUG | C:/Python/Python312/libs/python312_d.lib |
| PYTHON3_NUMPY_INCLUDE_DIRS | C:/Python/Python312/Lib/site-packages/numpy/core/include |
| PYTHON3_PACKAGES_PATH | C:/Python/Python312/Lib/site-packages |
| Qt5Concurrent_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Concurrent |
| Qt5Core_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Core |
| Qt5Gui_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Gui |
| Qt5OpenGL_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5OpenGL |
| Qt5Sql_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Sql |
| Qt5Test_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Test |
| Qt5Widgets_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Widgets |



| Name | Value |
|---|---|
| PYTHON3_LIBRARY | optimized;C:/Python/Python312/libs/python312.lib;debug;C:/Python/Python312/libs/python312_d.lib |
| PYTHON3_LIBRARY_DEBUG | C:/Python/Python312/libs/python312_d.lib |
| PYTHON3_NUMPY_INCLUDE_DIRS | C:/Python/Python312/Lib/site-packages/numpy/core/include |
| PYTHON3_PACKAGES_PATH | C:/Python/Python312/Lib/site-packages |
| Qt5Concurrent_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Concurrent |
| Qt5Core_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Core |
| Qt5Gui_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Gui |
| Qt5OpenGL_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5OpenGL |
| Qt5Sql_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Sql |
| Qt5Test_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Test |
| Qt5Widgets_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Widgets |
| Qt5_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5 |
| Qt6_DIR | Qt6_DIR-NOTFOUND |
| Tesseract_DIR | Tesseract_DIR-NOTFOUND |
| Tesseract_INCLUDE_DIR | Tesseract_INCLUDE_DIR-NOTFOUND |
| Tesseract_LIBRARY | Tesseract_LIBRARY-NOTFOUND |
| USE_WIN32_FILEIO | ☑ |
| VTK_DIR | C:/Program Files/VTK/lib/cmake/vtk-8.2 |
| WITH_1394 | ☑ |
| WITH_ADE | ☑ |
| WITH_ARITH_DEC | ☑ |
| WITH_ARITH_ENC | ☑ |
| WITH_AVIF | ☐ |
| WITH_CANN | ☐ |
| WITH_CLP | ☐ |
| WITH_CUBLAS | ☑ |
| WITH_CUDA | ☑ |
| WITH_CUDNN | ☑ |
| WITH_CUFFT | ☑ |
| WITH_DIRECTX | ☑ |
| WITH_DSHOW | ☑ |
| WITH_EIGEN | ☑ |
| WITH_FFMPEG | ☑ |
| WITH_FLATBUFFERS | ☑ |
| WITH_FREETYPE | ☐ |
| WITH_GDAL | ☐ |
| WITH_GDCM | ☐ |
| WITH_GSTREAMER | ☑ |
| WITH_HALIDE | ☐ |
| WITH_HPX | ☐ |
| WITH_IMGCODEC_HDR | ☑ |
| WITH_IMGCODEC_PFM | ☑ |
| WITH_IMGCODEC_PXM | ☑ |
| WITH_IMGCODEC_SUNRASTER | ☑ |
| WITH_IPP | ☑ |
| WITH_ITT | ☑ |
| WITH_JASPER | ☑ |
| WITH_JPEG | ☑ |
| WITH_LAPACK | ☑ |
| WITH_LIBREALSENSE | ☐ |
| WITH_MATLAB | ☐ |
| WITH_MFX | ☐ |



| Name | Value |
|---|---|
| WITH_ITT | ☑ |
| WITH_JASPER | ☑ |
| WITH_JPEG | ☑ |
| WITH_LAPACK | ☑ |
| WITH_LIBREALSENSE | ☐ |
| WITH_MATLAB | ☐ |
| WITH_MFX | ☐ |
| WITH_MSMF | ☑ |
| WITH_MSMF_DXVA | ☑ |
| WITH_NVCUVENC | ☑ |
| WITH_NVCUVID | ☑ |
| WITH_OAK | ☐ |
| WITH_OBSENSOR | ☑ |
| WITH_ONNX | ☐ |
| WITH_OPENCL | ☑ |
| WITH_OPENCLAMDBLAS | ☑ |
| WITH_OPENCLAMDFFT | ☑ |
| WITH_OPENCL_D3D11_NV | ☑ |
| WITH_OPENCL_SVM | ☐ |
| WITH_OPENEXR | ☑ |
| WITH_OPENGL | ☐ |
| WITH_OPENJPEG | ☑ |
| WITH_OPENMP | ☐ |
| WITH_OPENNI | ☐ |
| WITH_OPENNI2 | ☐ |
| WITH_OPENVINO | ☐ |
| WITH_OPENVX | ☐ |
| WITH_PLAIDML | ☐ |
| WITH_PNG | ☑ |
| WITH_PROTOBUF | ☑ |
| WITH_PVAPI | ☐ |
| WITH_QT | ☑ |
| WITH_QUIRC | ☑ |
| WITH_SPNG | ☐ |
| WITH_TBB | ☐ |
| WITH_TESSERACT | ☑ |
| WITH_TIFF | ☑ |
| WITH_TIMVX | ☐ |
| WITH_UEYE | ☐ |
| WITH_VTK | ☑ |
| WITH_VULKAN | ☐ |
| WITH_WEBNN | ☐ |
| WITH_WEBP | ☑ |
| WITH_WIN32UI | ☑ |
| WITH_XIMEA | ☐ |
| ccitt | ☑ |
| gflags_DIR | gflags_DIR-NOTFOUND |
| logluv | ☑ |
| lzw | ☑ |
| mdi | ☑ |
| next | ☑ |
| old-jpeg | ☐ |



CMake 3.27.6 - C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build

Where is the source code: C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0

Preset: <custom>

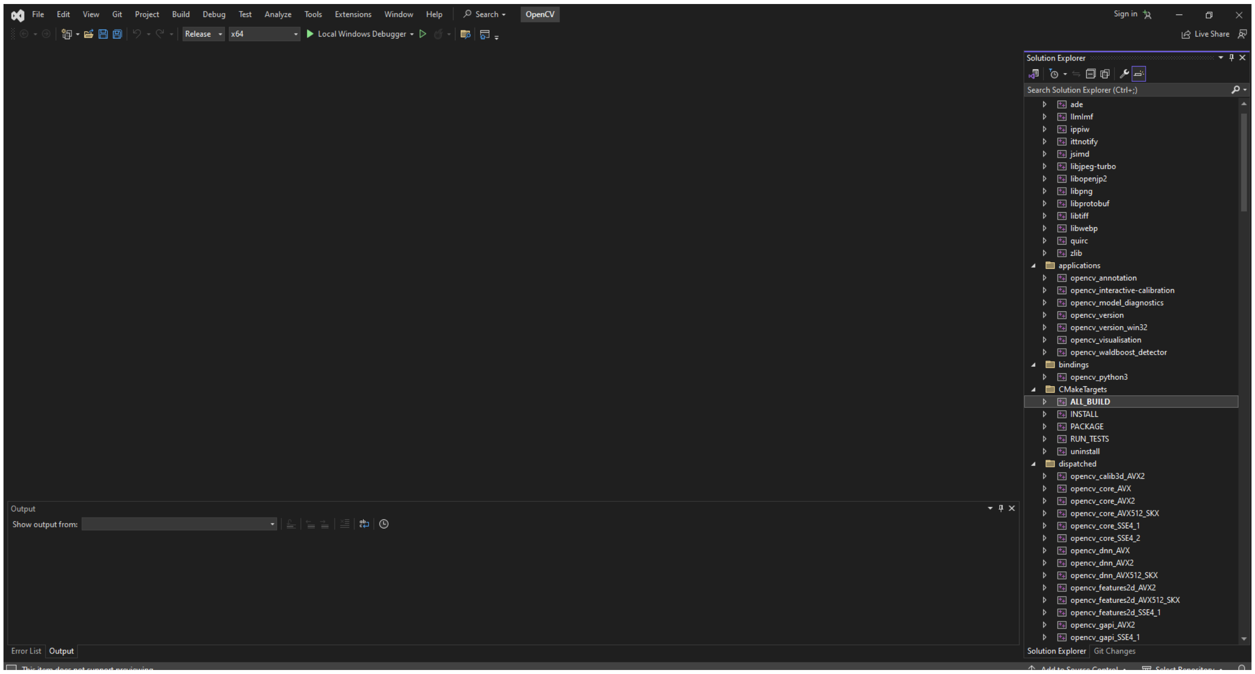
Where to build the binaries: C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build

Search:

| Name | Value |
|---|---|
| WITH_LAPACK | ☑ |
| WITH_LIBREALSENSE | ☐ |
| WITH_MATLAB | ☐ |
| WITH_MFX | ☐ |
| WITH_MSMF | ☑ |
| WITH_MSMF_DXVA | ☑ |
| WITH_NVCUVENC | ☑ |
| WITH_NVCUVID | ☑ |
| WITH_OAK | ☐ |
| WITH_OBSENSOR | ☑ |
| WITH_ONNX | ☐ |
| WITH_OPENCL | ☑ |
| WITH_OPENCLAMDBLAS | ☑ |
| WITH_OPENCLAMDFFT | ☑ |
| WITH_OPENCL_D3D11_NV | ☑ |
| WITH_OPENCL_SVM | ☐ |
| WITH_OPENEXR | ☑ |
| WITH_OPENGL | ☐ |
| WITH_OPENJPEG | ☑ |
| WITH_OPENMP | ☐ |
| WITH_OPENNI | ☐ |
| WITH_OPENNI2 | ☐ |
| WITH_OPENVINO | ☐ |
| WITH_OPENVX | ☐ |
| WITH_PLAIDML | ☐ |
| WITH_PNG | ☑ |
| WITH_PROTOBUF | ☑ |
| WITH_PVAPI | ☐ |
| WITH_QT | ☑ |
| WITH_QUIRC | ☑ |
| WITH_SPNG | ☐ |
| WITH_TBB | ☐ |
| WITH_TESSERACT | ☑ |
| WITH_TIFF | ☑ |
| WITH_TIMVX | ☐ |
| WITH_UEYE | ☐ |
| WITH_VTK | ☑ |
| WITH_VULKAN | ☐ |
| WITH_WEBNN | ☐ |
| WITH_WEBP | ☑ |
| WITH_WIN32UI | ☑ |
| WITH_XIMEA | ☐ |
| ccitt | ☑ |
| gflags_DIR | gflags_DIR-NOTFOUND |
| logluv | ☑ |
| lzw | ☑ |
| mdi | ☑ |
| next | ☑ |
| old-jpeg | ☐ |
| packbits | ☑ |
| thunder | ☑ |

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

Current Generator: Visual Studio 17 2022



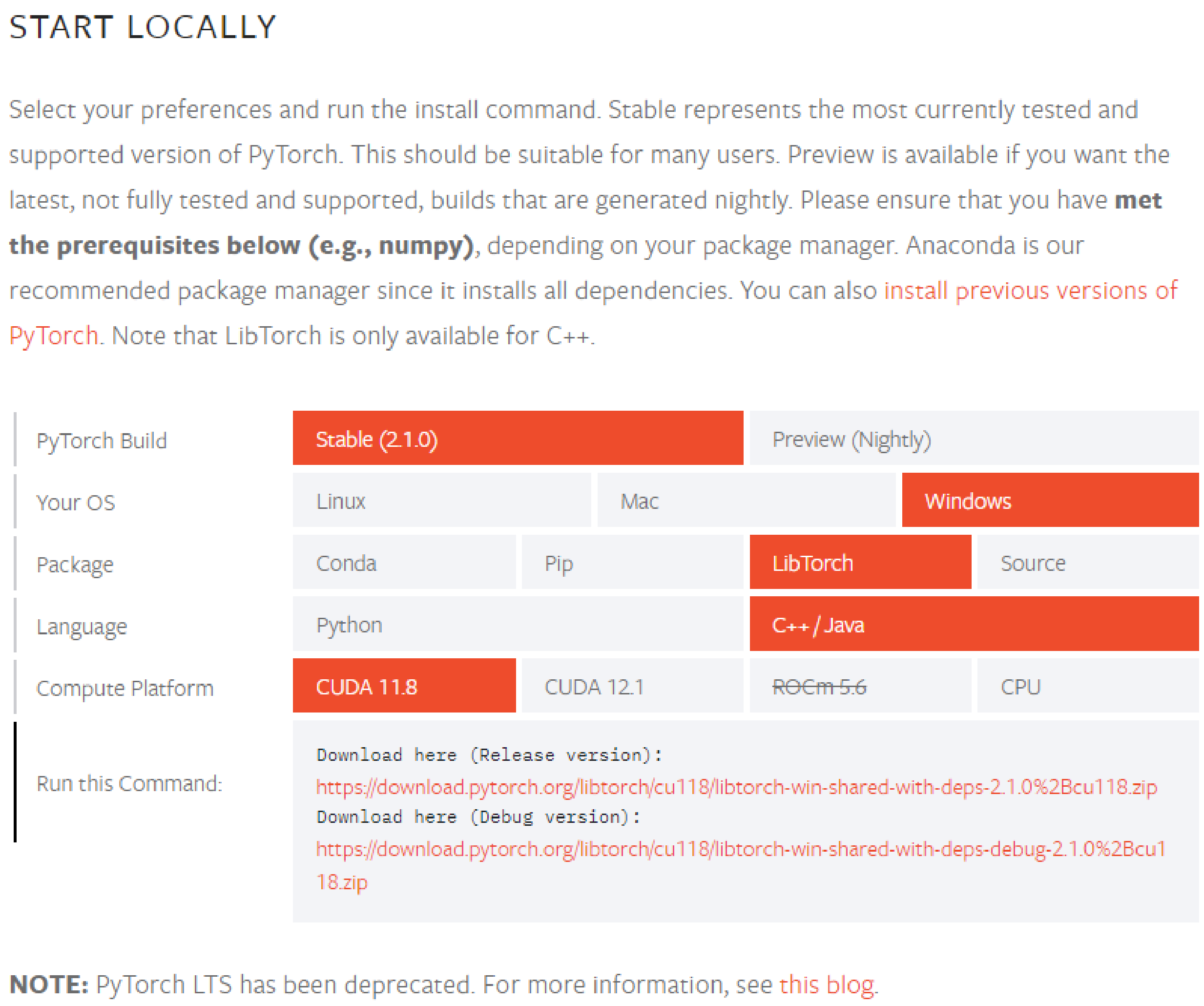
Release | x64

Solution Explorer

- ade
- IlmImf
- ippiw
- ittnotify
- jsimd
- libjpeg-turbo
- libopenjp2
- libpng
- libprotobuf
- libtiff
- libwebp
- quirc
- zlib
- applications
  - opencv_annotation
  - opencv_interactive-calibration
  - opencv_model_diagnostics
  - opencv_version
  - opencv_version_win32
  - opencv_visualisation
  - opencv_waldboost_detector
- bindings
  - opencv_python3
- CMakeTargets
  - **ALL_BUILD**
  - INSTALL
  - PACKAGE
  - RUN_TESTS
  - uninstall
- dispatched
  - opencv_calib3d_AVX2
  - opencv_core_AVX
  - opencv_core_AVX2
  - opencv_core_AVX512_SKX
  - opencv_core_SSE4_1
  - opencv_core_SSE4_2
  - opencv_dnn_AVX
  - opencv_dnn_AVX2
  - opencv_dnn_AVX512_SKX
  - opencv_features2d_AVX2
  - opencv_features2d_AVX512_SKX
  - opencv_features2d_SSE4_1
  - opencv_gapi_AVX2
  - opencv_gapi_SSE4_1

# 9 YAML CPP and LibTorch installation

You will also need other libraries. One is YAML CPP which is used for writing code outputs, you can find it attached as YAML_CPP0.6.zip. You will also need the libtorch (pythorch) library for your corresponding Cuda version. I am using ‘libtorch-win-shared-with-deps-2.1.0+cu118.zip’. you can download from https://pytorch.org/get-started/locally/. These two zip files should only be extracted, and do not need compiling.



START LOCALLY

Select your preferences and run the install command. Stable represents the most currently tested and supported version of PyTorch. This should be suitable for many users. Preview is available if you want the latest, not fully tested and supported, builds that are generated nightly. Please ensure that you have **met the prerequisites below (e.g., numpy)**, depending on your package manager. Anaconda is our recommended package manager since it installs all dependencies. You can also install previous versions of PyTorch. Note that LibTorch is only available for C++.

PyTorch Build: [✓ Stable (2.1.0)] Preview (Nightly)

Your OS: Linux Mac [✓ Windows]

Package: Conda Pip [✓ LibTorch] Source

Language: Python [✓ C++ / Java]

Compute Platform: [✓ CUDA 11.8] CUDA 12.1 ~~ROCm 5.6~~ CPU

Run this Command:

```
Download here (Release version):
https://download.pytorch.org/libtorch/cu118/libtorch-win-shared-with-deps-2.1.0%2Bcu118.zip
Download here (Debug version):
https://download.pytorch.org/libtorch/cu118/libtorch-win-shared-with-deps-debug-2.1.0%2Bcu118.zip
```

**NOTE:** PyTorch LTS has been deprecated. For more information, see this blog.

# 10 OptiTrack Camera SDK Installation

You need to install OptiTrack Camera SDK. I am using version 2.3.1 and the file name is 'OptiTrack_Camera_SDK_2.3.1_Final.exe'. My installation path is: C:\Program Files (x86)\OptiTrack\Camera SDK

# 11 Compiling the main VPTV code

Now you have installed all the required libraries, you need to build the VPTV software. Currently, the code on GitHub is not working as expected, and there is another version of the code available with some functionalities commented and disabled. Here we will build the commented code. You can find the commented code attached to this file. Follow the steps below:

First, you need to make some changes in your cmakelist.txt file located in the project main folder. Mine is at: C:\mahdi\VPTV_VS2022\CMakeLists.txt

1- I added some lines regarding the cmake Policy CMP0146 changes, which was shown as a warning when building the solution file. This is dependent on your opencv version, and you may not get the same warning. If you get the warning below:

```
CMake Warning (dev) at packages/opencv-4.8.0/build/OpenCVConfig.cmake:86
(find_package):
  Policy CMP0146 is not set: The FindCUDA module is removed.  Run "cmake
  --help-policy CMP0146" for policy details.  Use the cmake_policy command
to set the policy and suppress this warning.
Call Stack (most recent call first):
  packages/opencv-4.8.0/build/OpenCVConfig.cmake:108 (find_host_package)
  CMakeLists.txt:12 (find_package)
This warning is for project developers.  Use -Wno-dev to suppress it.
```

So I added the following lines in the cmakelists.txt on the second line after the project name:

```
if (POLICY CMP0146)

      cmake_policy(SET CMP0146 OLD)

endif()
```

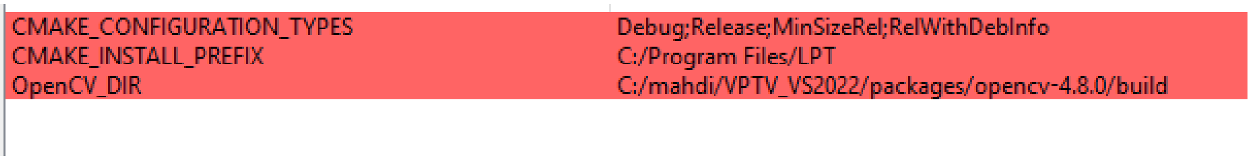
you can see my CMake logs before and after these modifications, if necessary.

2- You can adjust your opencv and cuda versions at the lines find_package( OpenCV 4.8.0 REQUIRED) and find_package( CUDA 11.8 )

3- You need to change the boost library location and version on the lines ‘set(BOOST_ROOT C:/mahdi/VPTV_VS2022/packages/boost_1_83_0)’ and ‘set(Boost_ADDITIONAL_VERSIONS "1.83" "1.83.0")’ and ‘find_package( Boost 1.83.0 COMPONENTS thread chrono random date_time system REQUIRED )’. When changing the boost root dir, you need to change ‘/’ to ‘\’ if you copy the directory from Windows explorer.

After modification of your cmakelists.txt file, you can start building steps as follows:

1. Open cmake and navigate to your project directory, click configure as shown, and click finish
2. You get some error because it cannot find the open CV directory, so manually do it and point to your opencv build directory as shown. 5/24/2024: change config types to Release.



| | |
|---|---|
| CMAKE_CONFIGURATION_TYPES | Debug;Release;MinSizeRel;RelWithDebInfo |
| CMAKE_INSTALL_PREFIX | C:/Program Files/LPT |
| OpenCV_DIR | C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build |

3. You need to manually set the directory of other libraries as:

- TORCH_LIBRARY: C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/lib/torch.lib
- YAMLCPP_HAVE_H: TRUE
- YAMLCPP_H_INCLUDE_DIR: C:/mahdi/VPTV_VS2022/packages/YAML_CPP0.6/YAML_CPP 0.6/include
- YAMLCPP_LIBRARY: C:/mahdi/VPTV_VS2022/packages/YAML_CPP0.6/YAML_CPP 0.6/lib/yaml-cpp.lib

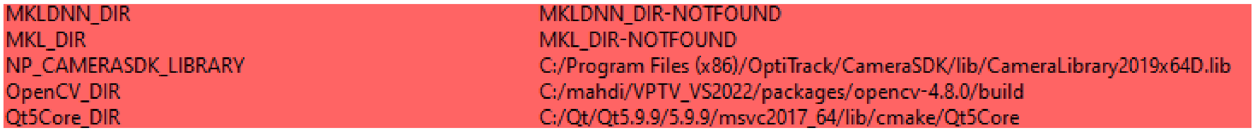
4. Click Configure again and make sure all variables are similar to the screenshot provided below.
5. Click generate and open project in VS 2022 as admin
6. You may get a prompt to reload, so reload all
7. After the successful build of all packages, go to the build folder. Mine is at C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo. Then you need to copy and paste many local files here for the program to work.
8. 'CameraLibrary2015x64D.dll' from C:\Program Files (x86)\OptiTrack\Camera SDK\lib to C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo
9. All *.dll files (application extension) from C:\Qt\Qt5.9.9\5.9.9\msvc2017_64\bin to C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo
10. All *.dll files (application extension) from C:\Program Files\VTK\bin to C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo
11. All *.dll files (application extension) from C:\mahdi\VPTV_VS2022\packages\opencv-4.8.0\build\install\x64\vc17\bin to C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo
12. 'yaml-cpp.dll' from C:\mahdi\VPTV_VS2022\packages\YAML_CPP0.6\YAML_CPP 0.6\bin to C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo
13. All *.dll files (application extension) from C:\mahdi\VPTV_VS2022\packages\libtorch-win-shared-with-deps-2.1.0+cu118\libtorch\lib to C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo
14. In the build directory (C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo ), you can open the file 'app-LPT_Optitrack.exe' to connect to real cameras, or 'app-LPT_Virtual.exe' to see a virtual demonstration window.
15. The software might not need all *.dll files copied above, maybe they can be filtered in the future.

Update on 5/7/2024 at 12.22 am: add this to windows environment variables so cmake can find open cv

$env:OpenCV_DIR = "C:\mahdi\VPTV_VS2022\packages\opencv-4.8.0\build"

Update 5/24/2024: SDK changed to latest version 3.1.0. Exe of installer included, and it is installed in 'C:\Program Files (x86)\OptiTrack\CameraSDK' while 2.3 was installed in 'C:\Program Files (x86)\OptiTrack\Camera SDK' note the space. The environment variable is changed by the installer itself. The only change needed is

The image after change to change the contents of the file FindNP_CAMERASDK.cmake in dir C:\mahdi\VPTV_VS2022\scripts\cmake.



| | |
|---|---|
| MKLDNN_DIR | MKLDNN_DIR-NOTFOUND |
| MKL_DIR | MKL_DIR-NOTFOUND |
| NP_CAMERASDK_LIBRARY | C:/Program Files (x86)/OptiTrack/CameraSDK/lib/CameraLibrary2019x64D.lib |
| OpenCV_DIR | C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build |
| Qt5Core_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Core |

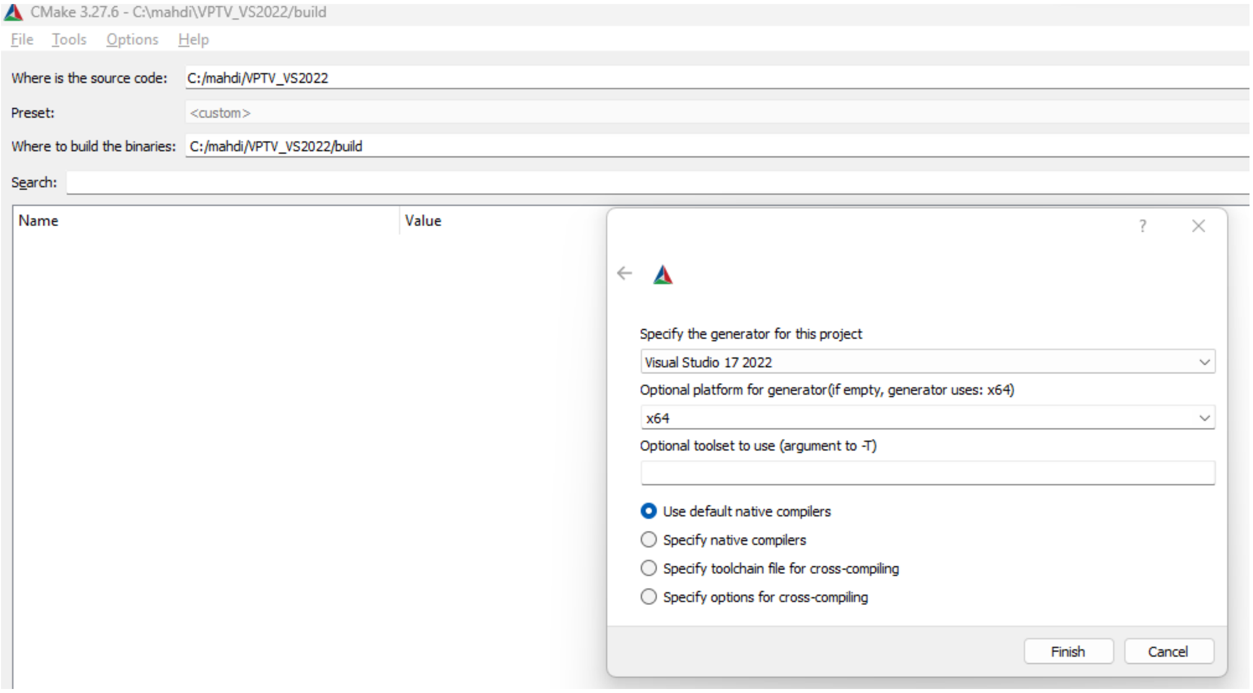
New contents:

```
# Find Natural Point Camera SDK for Windows
# This will define:
#  NP_CAMERASDK_FOUND - System has the Camera SDK
#  NP_CAMERASDK_INCLUDE_DIRS - The Camera SDK include directories
#  NP_CAMERASDK_LIBRARY_DIRS - The Camera SDK library directories
#  NP_CAMERASDK_LIBRARIES - The libraries needed to use the Camera SDK
#  NP_CAMERASDK_DEFINITIONS - Compiler switches

set(NP_CAMERASDK_INCLUDE_DIR $ENV{NP_CAMERASDK}/include)

# Determine bitness of the environment
if(CMAKE_SIZEOF_VOID_P EQUAL 8)
   message(STATUS "64-bit environment")
   # Update library name for 64-bit
   find_library(NP_CAMERASDK_LIBRARY
      NAMES CameraLibrary2019x64D.lib
      HINTS $ENV{NP_CAMERASDK}/lib)
   add_definitions(-DWIN64)
else()
   message(STATUS "32-bit environment")
   # Update or confirm library name for 32-bit if available
   find_library(NP_CAMERASDK_LIBRARY
      NAMES CameraLibrary2019.lib  # Adjust name as needed
      HINTS $ENV{NP_CAMERASDK}/lib)
   add_definitions(-DWIN32)
endif()

set(NP_CAMERASDK_LIBRARIES ${NP_CAMERASDK_LIBRARY})
set(NP_CAMERASDK_LIBRARY_DIRS $ENV{NP_CAMERASDK}/lib)
set(NP_CAMERASDK_INCLUDE_DIRS ${NP_CAMERASDK_INCLUDE_DIR})

add_definitions(-DCAMERALIBRARY_IMPORTS)
add_definitions(-DUSE_NP_CAMERASDK)
link_directories(${NP_CAMERASDK_LIBRARY_DIRS})
include_directories(${NP_CAMERASDK_INCLUDE_DIRS})

message(STATUS "NP Camera SDK Root: " $ENV{NP_CAMERASDK})
message(STATUS "Include: " ${NP_CAMERASDK_INCLUDE_DIRS})
message(STATUS "Library Dir: " ${NP_CAMERASDK_LIBRARY_DIRS})
message(STATUS "Library: " ${NP_CAMERASDK_LIBRARIES})

include(FindPackageHandleStandardArgs)
find_package_handle_standard_args(NP_CAMERASDK DEFAULT_MSG
                                  NP_CAMERASDK_LIBRARY NP_CAMERASDK_INCLUDE_DIR)

mark_as_advanced(NP_CAMERASDK_INCLUDE_DIR NP_CAMERASDK_LIBRARY)
```

update2: I figured out the 2019D.dll is for debug mode which is wrong to use in our case, we need to use 2019s.dll. mody the cmake script as follows:

```
# Find Natural Point Camera SDK for Windows
set(NP_CAMERASDK_INCLUDE_DIR $ENV{NP_CAMERASDK}/include)

# Assuming 64-bit environment as default
message(STATUS "Configuring for 64-bit environment")
find_library(NP_CAMERASDK_LIBRARY
   NAMES CameraLibrary2019x64S.lib
   HINTS $ENV{NP_CAMERASDK}/lib)
```

```
add_definitions(-DWIN64)

set(NP_CAMERASDK_LIBRARIES ${NP_CAMERASDK_LIBRARY})
set(NP_CAMERASDK_LIBRARY_DIRS $ENV{NP_CAMERASDK}/lib)
set(NP_CAMERASDK_INCLUDE_DIRS ${NP_CAMERASDK_INCLUDE_DIR})

add_definitions(-DCAMERALIBRARY_IMPORTS)
add_definitions(-DUSE_NP_CAMERASDK)
link_directories(${NP_CAMERASDK_LIBRARY_DIRS})
include_directories(${NP_CAMERASDK_INCLUDE_DIRS})

message(STATUS "NP Camera SDK Root: " $ENV{NP_CAMERASDK})
message(STATUS "Include: " ${NP_CAMERASDK_INCLUDE_DIRS})
message(STATUS "Library Dir: " ${NP_CAMERASDK_LIBRARY_DIRS})
message(STATUS "Library: " ${NP_CAMERASDK_LIBRARIES})

include(FindPackageHandleStandardArgs)
find_package_handle_standard_args(NP_CAMERASDK DEFAULT_MSG
                          NP_CAMERASDK_LIBRARY NP_CAMERASDK_INCLUDE_DIR)
```



Where is the source code: C:/mahdi/VPTV_VS2022

Preset: <custom>

Where to build the binaries: C:/mahdi/VPTV_VS2022/build

Search:

| Name | Value |
|---|---|

Specify the generator for this project: Visual Studio 17 2022

Optional platform for generator(if empty, generator uses: x64): x64

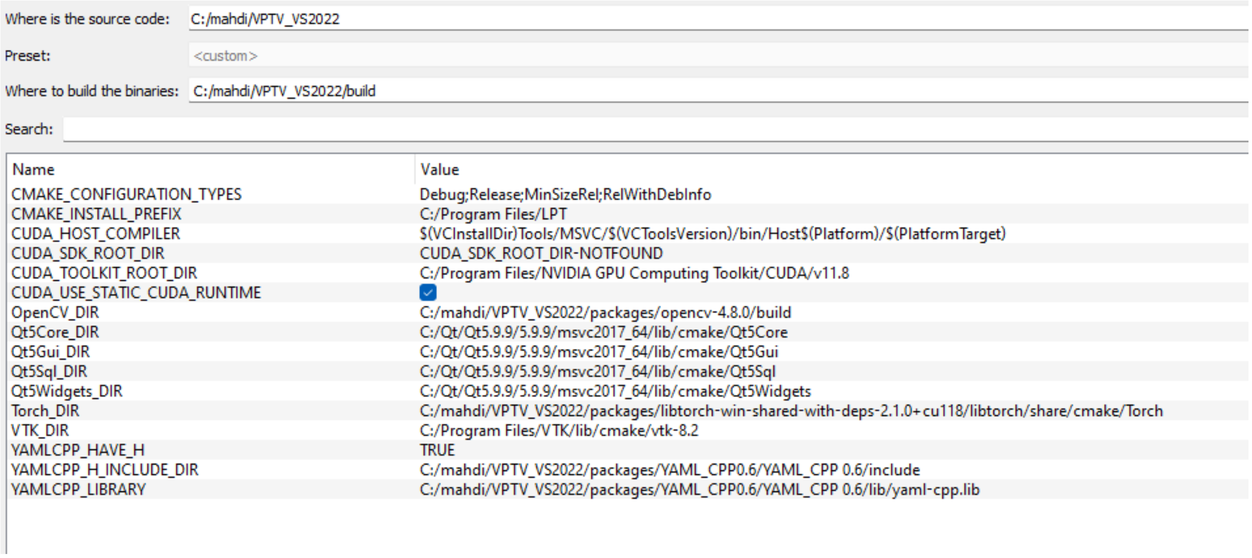
Optional toolset to use (argument to -T):

● Use default native compilers

○ Specify native compilers

○ Specify toolchain file for cross-compiling

○ Specify options for cross-compiling



Where is the source code: C:/mahdi/VPTV_VS2022

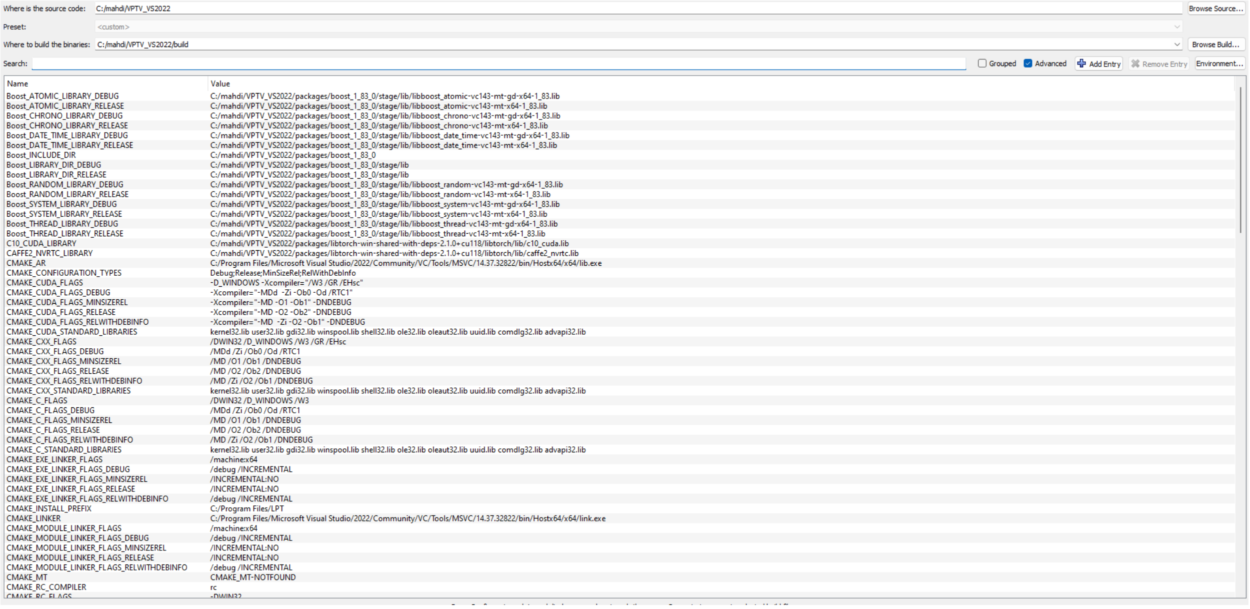
Preset: <custom>

Where to build the binaries: C:/mahdi/VPTV_VS2022/build

Search:

| Name | Value |
|---|---|
| CMAKE_CONFIGURATION_TYPES | Debug;Release;MinSizeRel;RelWithDebInfo |
| CMAKE_INSTALL_PREFIX | C:/Program Files/LPT |
| CUDA_HOST_COMPILER | $(VCInstallDir)Tools/MSVC/$(VCToolsVersion)/bin/Host$(Platform)/$(PlatformTarget) |
| CUDA_SDK_ROOT_DIR | CUDA_SDK_ROOT_DIR-NOTFOUND |
| CUDA_TOOLKIT_ROOT_DIR | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8 |
| CUDA_USE_STATIC_CUDA_RUNTIME | ☑ |
| OpenCV_DIR | C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build |
| Qt5Core_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Core |
| Qt5Gui_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Gui |
| Qt5Sql_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Sql |
| Qt5Widgets_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Widgets |
| Torch_DIR | C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/share/cmake/Torch |
| VTK_DIR | C:/Program Files/VTK/lib/cmake/vtk-8.2 |
| YAMLCPP_HAVE_H | TRUE |
| YAMLCPP_H_INCLUDE_DIR | C:/mahdi/VPTV_VS2022/packages/YAML_CPP0.6/YAML_CPP 0.6/include |
| YAMLCPP_LIBRARY | C:/mahdi/VPTV_VS2022/packages/YAML_CPP0.6/YAML_CPP 0.6/lib/yaml-cpp.lib |

After turning on advanced:



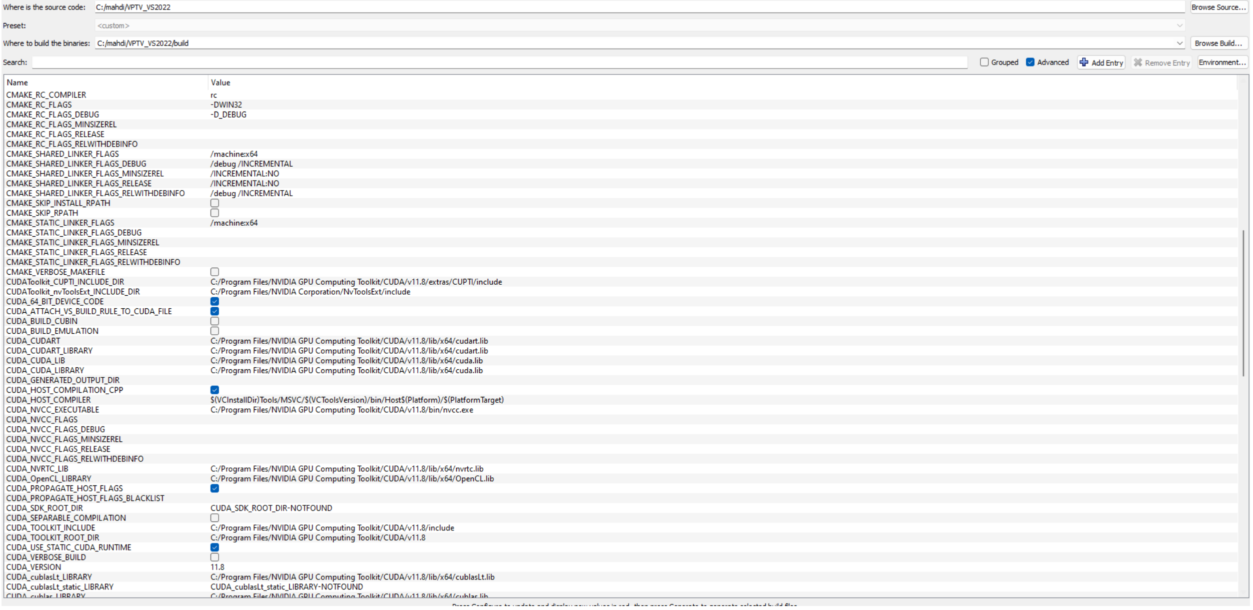
Where is the source code: C:/mahdi/VPTV_VS2022

Preset: <custom>

Where to build the binaries: C:/mahdi/VPTV_VS2022/build

Search:

| Name | Value |
|---|---|
| Boost_ATOMIC_LIBRARY_DEBUG | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_atomic-vc143-mt-gd-x64-1_83.lib |
| Boost_ATOMIC_LIBRARY_RELEASE | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_atomic-vc143-mt-x64-1_83.lib |
| Boost_CHRONO_LIBRARY_DEBUG | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_chrono-vc143-mt-gd-x64-1_83.lib |
| Boost_CHRONO_LIBRARY_RELEASE | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_chrono-vc143-mt-x64-1_83.lib |
| Boost_DATE_TIME_LIBRARY_DEBUG | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_date_time-vc143-mt-gd-x64-1_83.lib |
| Boost_DATE_TIME_LIBRARY_RELEASE | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_date_time-vc143-mt-x64-1_83.lib |
| Boost_INCLUDE_DIR | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0 |
| Boost_LIBRARY_DIR_DEBUG | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib |
| Boost_LIBRARY_DIR_RELEASE | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib |
| Boost_RANDOM_LIBRARY_DEBUG | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_random-vc143-mt-gd-x64-1_83.lib |
| Boost_RANDOM_LIBRARY_RELEASE | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_random-vc143-mt-x64-1_83.lib |
| Boost_SYSTEM_LIBRARY_DEBUG | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_system-vc143-mt-gd-x64-1_83.lib |
| Boost_SYSTEM_LIBRARY_RELEASE | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_system-vc143-mt-x64-1_83.lib |
| Boost_THREAD_LIBRARY_DEBUG | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_thread-vc143-mt-gd-x64-1_83.lib |
| Boost_THREAD_LIBRARY_RELEASE | C:/mahdi/VPTV_VS2022/packages/boost_1_83_0/stage/lib/libboost_thread-vc143-mt-x64-1_83.lib |
| C10_CUDA_LIBRARY | C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/lib/c10_cuda.lib |
| CAFFE2_NVRTC_LIBRARY | C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/lib/caffe2_nvrtc.lib |
| CMAKE_AR | C:/Program Files/Microsoft Visual Studio/2022/Community/VC/Tools/MSVC/14.37.32822/bin/Hostx64/x64/lib.exe |
| CMAKE_CONFIGURATION_TYPES | Debug;Release;MinSizeRel;RelWithDebInfo |
| CMAKE_CUDA_FLAGS | -D_WINDOWS -Xcompiler="/W3 /GR /EHsc" |
| CMAKE_CUDA_FLAGS_DEBUG | -Xcompiler="-MDd  -Zi -Ob0 -Od /RTC1" |
| CMAKE_CUDA_FLAGS_MINSIZEREL | -Xcompiler="-MD -O1 -Ob1" -DNDEBUG |
| CMAKE_CUDA_FLAGS_RELEASE | -Xcompiler="-MD -O2 -Ob2" -DNDEBUG |
| CMAKE_CUDA_FLAGS_RELWITHDEBINFO | -Xcompiler="-MD  -Zi -O2 -Ob1" -DNDEBUG |
| CMAKE_CUDA_STANDARD_LIBRARIES | kernel32.lib user32.lib gdi32.lib winspool.lib shell32.lib ole32.lib oleaut32.lib uuid.lib comdlg32.lib advapi32.lib |
| CMAKE_CXX_FLAGS | /DWIN32 /D_WINDOWS /W3 /GR /EHsc |
| CMAKE_CXX_FLAGS_DEBUG | /MDd /Zi /Ob0 /Od /RTC1 |
| CMAKE_CXX_FLAGS_MINSIZEREL | /MD /O1 /Ob1 /DNDEBUG |
| CMAKE_CXX_FLAGS_RELEASE | /MD /O2 /Ob2 /DNDEBUG |
| CMAKE_CXX_FLAGS_RELWITHDEBINFO | /MD /Zi /O2 /Ob1 /DNDEBUG |
| CMAKE_CXX_STANDARD_LIBRARIES | kernel32.lib user32.lib gdi32.lib winspool.lib shell32.lib ole32.lib oleaut32.lib uuid.lib comdlg32.lib advapi32.lib |
| CMAKE_C_FLAGS | /DWIN32 /D_WINDOWS /W3 |
| CMAKE_C_FLAGS_DEBUG | /MDd /Zi /Ob0 /Od /RTC1 |
| CMAKE_C_FLAGS_MINSIZEREL | /MD /O1 /Ob1 /DNDEBUG |
| CMAKE_C_FLAGS_RELEASE | /MD /O2 /Ob2 /DNDEBUG |
| CMAKE_C_FLAGS_RELWITHDEBINFO | /MD /Zi /O2 /Ob1 /DNDEBUG |
| CMAKE_C_STANDARD_LIBRARIES | kernel32.lib user32.lib gdi32.lib winspool.lib shell32.lib ole32.lib oleaut32.lib uuid.lib comdlg32.lib advapi32.lib |
| CMAKE_EXE_LINKER_FLAGS | /machine:x64 |
| CMAKE_EXE_LINKER_FLAGS_DEBUG | /debug /INCREMENTAL |
| CMAKE_EXE_LINKER_FLAGS_MINSIZEREL | /INCREMENTAL:NO |
| CMAKE_EXE_LINKER_FLAGS_RELEASE | /INCREMENTAL:NO |
| CMAKE_EXE_LINKER_FLAGS_RELWITHDEBINFO | /debug /INCREMENTAL |
| CMAKE_INSTALL_PREFIX | C:/Program Files/LPT |
| CMAKE_LINKER | C:/Program Files/Microsoft Visual Studio/2022/Community/VC/Tools/MSVC/14.37.32822/bin/Hostx64/x64/link.exe |
| CMAKE_MODULE_LINKER_FLAGS | /machine:x64 |
| CMAKE_MODULE_LINKER_FLAGS_DEBUG | /debug /INCREMENTAL |
| CMAKE_MODULE_LINKER_FLAGS_MINSIZEREL | /INCREMENTAL:NO |
| CMAKE_MODULE_LINKER_FLAGS_RELEASE | /INCREMENTAL:NO |
| CMAKE_MODULE_LINKER_FLAGS_RELWITHDEBINFO | /debug /INCREMENTAL |
| CMAKE_MT | CMAKE_MT-NOTFOUND |
| CMAKE_RC_COMPILER | rc |
| CMAKE_RC_FLAGS | -DWIN32 |



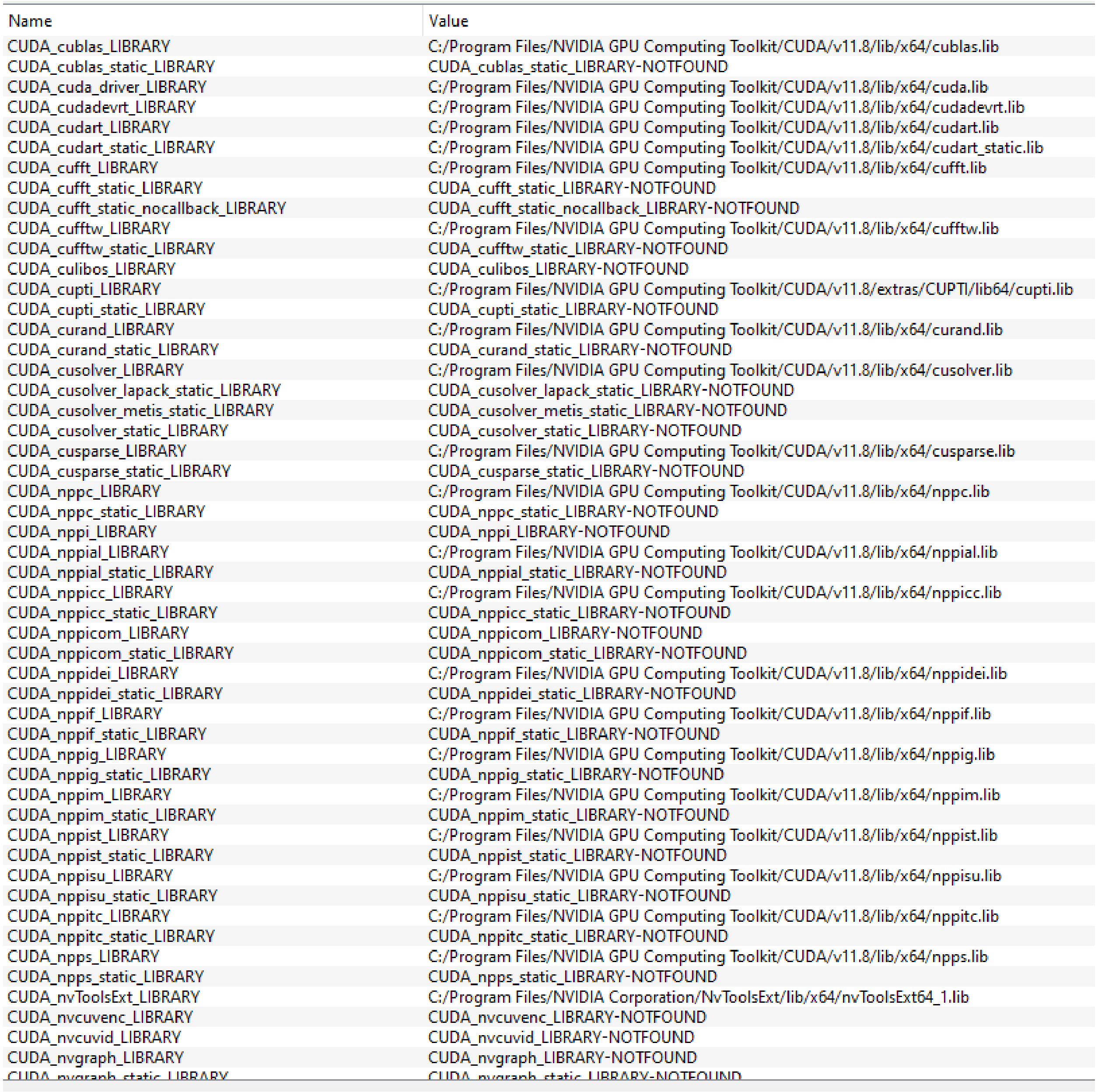
Where is the source code: C:/mahdi/VPTV_VS2022

Preset: <custom>

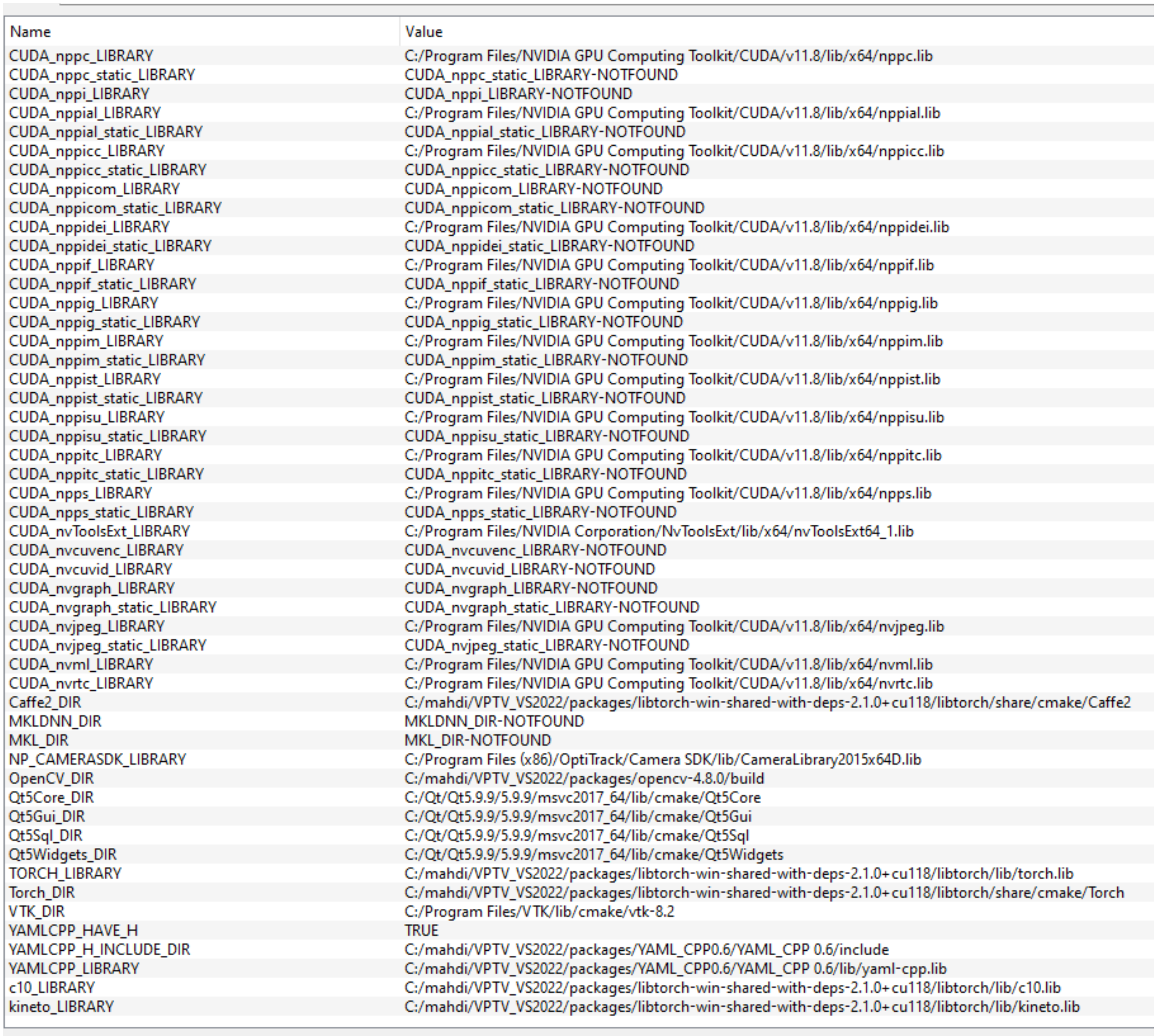
Where to build the binaries: C:/mahdi/VPTV_VS2022/build

Search:

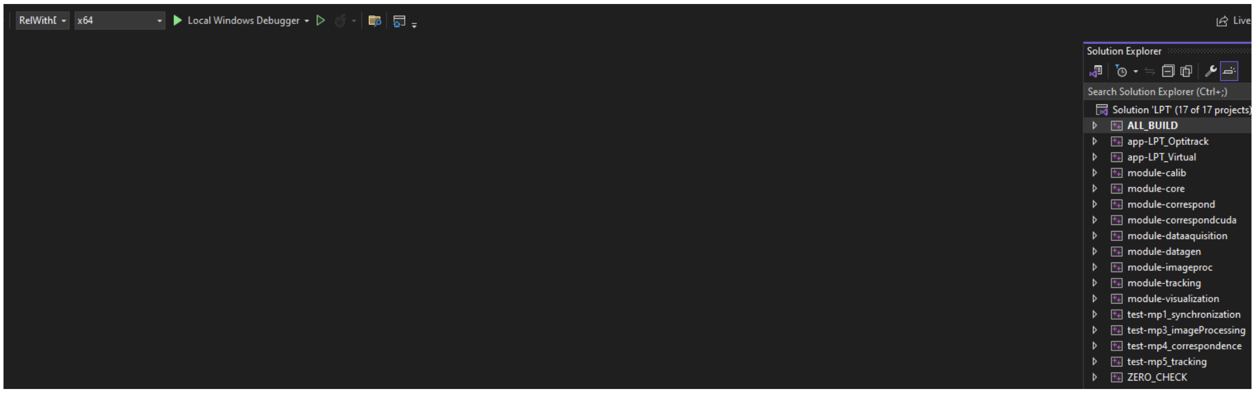
| Name | Value |
|---|---|
| CMAKE_RC_COMPILER | rc |
| CMAKE_RC_FLAGS | -DWIN32 |
| CMAKE_RC_FLAGS_DEBUG | -D_DEBUG |
| CMAKE_RC_FLAGS_MINSIZEREL | |
| CMAKE_RC_FLAGS_RELEASE | |
| CMAKE_RC_FLAGS_RELWITHDEBINFO | |
| CMAKE_SHARED_LINKER_FLAGS | /machine:x64 |
| CMAKE_SHARED_LINKER_FLAGS_DEBUG | /debug /INCREMENTAL |
| CMAKE_SHARED_LINKER_FLAGS_MINSIZEREL | /INCREMENTAL:NO |
| CMAKE_SHARED_LINKER_FLAGS_RELEASE | /INCREMENTAL:NO |
| CMAKE_SHARED_LINKER_FLAGS_RELWITHDEBINFO | /debug /INCREMENTAL |
| CMAKE_SKIP_INSTALL_RPATH | □ |
| CMAKE_SKIP_RPATH | □ |
| CMAKE_STATIC_LINKER_FLAGS | /machine:x64 |
| CMAKE_STATIC_LINKER_FLAGS_DEBUG | |
| CMAKE_STATIC_LINKER_FLAGS_MINSIZEREL | |
| CMAKE_STATIC_LINKER_FLAGS_RELEASE | |
| CMAKE_STATIC_LINKER_FLAGS_RELWITHDEBINFO | |
| CMAKE_VERBOSE_MAKEFILE | □ |
| CUDAToolkit_CUPTI_INCLUDE_DIR | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/extras/CUPTI/include |
| CUDAToolkit_nvToolsExt_INCLUDE_DIR | C:/Program Files/NVIDIA Corporation/NvToolsExt/include |
| CUDA_64_BIT_DEVICE_CODE | ☑ |
| CUDA_ATTACH_VS_BUILD_RULE_TO_CUDA_FILE | ☑ |
| CUDA_BUILD_CUBIN | □ |
| CUDA_BUILD_EMULATION | □ |
| CUDA_CUDART | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cudart.lib |
| CUDA_CUDART_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cudart.lib |
| CUDA_CUDA_LIB | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cuda.lib |
| CUDA_CUDA_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cuda.lib |
| CUDA_GENERATED_OUTPUT_DIR | |
| CUDA_HOST_COMPILATION_CPP | ☑ |
| CUDA_HOST_COMPILER | $(VCInstallDir)Tools/MSVC/$(VCToolsVersion)/bin/Host$(Platform)/$(PlatformTarget) |
| CUDA_NVCC_EXECUTABLE | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/bin/nvcc.exe |
| CUDA_NVCC_FLAGS | |
| CUDA_NVCC_FLAGS_DEBUG | |
| CUDA_NVCC_FLAGS_MINSIZEREL | |
| CUDA_NVCC_FLAGS_RELEASE | |
| CUDA_NVCC_FLAGS_RELWITHDEBINFO | |
| CUDA_NVRTC_LIB | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nvrtc.lib |
| CUDA_OpenCL_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/OpenCL.lib |
| CUDA_PROPAGATE_HOST_FLAGS | ☑ |
| CUDA_PROPAGATE_HOST_FLAGS_BLACKLIST | |
| CUDA_SDK_ROOT_DIR | CUDA_SDK_ROOT_DIR-NOTFOUND |
| CUDA_SEPARABLE_COMPILATION | □ |
| CUDA_TOOLKIT_INCLUDE | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/include |
| CUDA_TOOLKIT_ROOT_DIR | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8 |
| CUDA_USE_STATIC_CUDA_RUNTIME | ☑ |
| CUDA_VERBOSE_BUILD | □ |
| CUDA_VERSION | 11.8 |
| CUDA_cublasLt_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cublasLt.lib |
| CUDA_cublasLt_static_LIBRARY | CUDA_cublasLt_static_LIBRARY-NOTFOUND |
| CUDA_cublas_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cublas.lib |



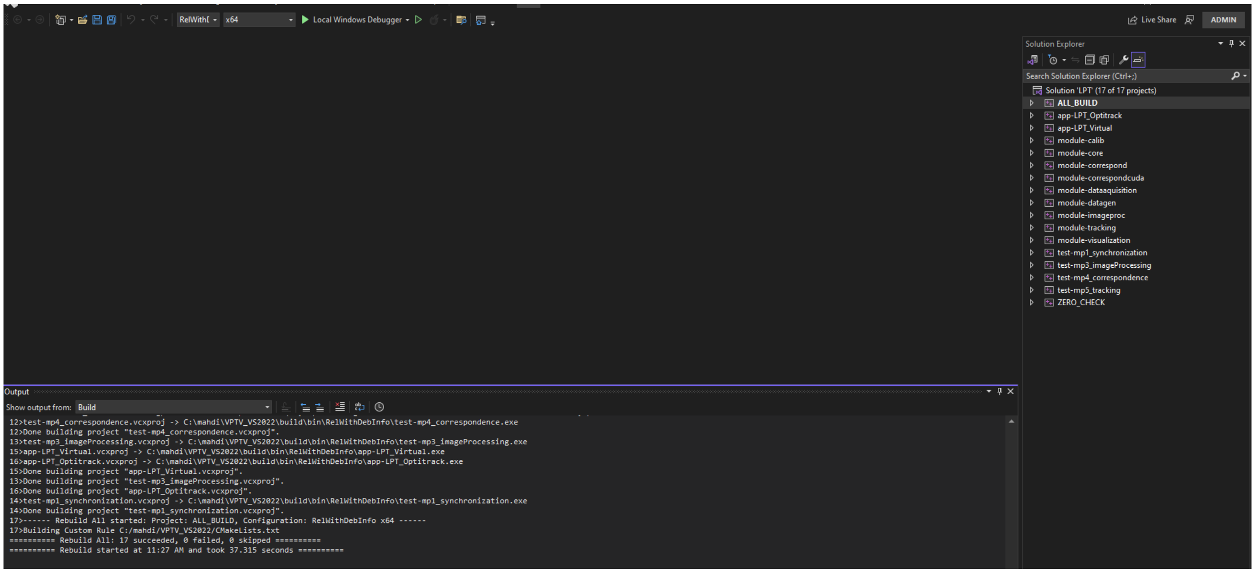
| Name | Value |
|---|---|
| CUDA_cublas_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cublas.lib |
| CUDA_cublas_static_LIBRARY | CUDA_cublas_static_LIBRARY-NOTFOUND |
| CUDA_cuda_driver_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cuda.lib |
| CUDA_cudadevrt_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cudadevrt.lib |
| CUDA_cudart_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cudart.lib |
| CUDA_cudart_static_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cudart_static.lib |
| CUDA_cufft_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cufft.lib |
| CUDA_cufft_static_LIBRARY | CUDA_cufft_static_LIBRARY-NOTFOUND |
| CUDA_cufft_static_nocallback_LIBRARY | CUDA_cufft_static_nocallback_LIBRARY-NOTFOUND |
| CUDA_cufftw_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cufftw.lib |
| CUDA_cufftw_static_LIBRARY | CUDA_cufftw_static_LIBRARY-NOTFOUND |
| CUDA_culibos_LIBRARY | CUDA_culibos_LIBRARY-NOTFOUND |
| CUDA_cupti_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/extras/CUPTI/lib64/cupti.lib |
| CUDA_cupti_static_LIBRARY | CUDA_cupti_static_LIBRARY-NOTFOUND |
| CUDA_curand_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/curand.lib |
| CUDA_curand_static_LIBRARY | CUDA_curand_static_LIBRARY-NOTFOUND |
| CUDA_cusolver_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cusolver.lib |
| CUDA_cusolver_lapack_static_LIBRARY | CUDA_cusolver_lapack_static_LIBRARY-NOTFOUND |
| CUDA_cusolver_metis_static_LIBRARY | CUDA_cusolver_metis_static_LIBRARY-NOTFOUND |
| CUDA_cusolver_static_LIBRARY | CUDA_cusolver_static_LIBRARY-NOTFOUND |
| CUDA_cusparse_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/cusparse.lib |
| CUDA_cusparse_static_LIBRARY | CUDA_cusparse_static_LIBRARY-NOTFOUND |
| CUDA_nppc_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppc.lib |
| CUDA_nppc_static_LIBRARY | CUDA_nppc_static_LIBRARY-NOTFOUND |
| CUDA_nppi_LIBRARY | CUDA_nppi_LIBRARY-NOTFOUND |
| CUDA_nppial_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppial.lib |
| CUDA_nppial_static_LIBRARY | CUDA_nppial_static_LIBRARY-NOTFOUND |
| CUDA_nppicc_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppicc.lib |
| CUDA_nppicc_static_LIBRARY | CUDA_nppicc_static_LIBRARY-NOTFOUND |
| CUDA_nppicom_LIBRARY | CUDA_nppicom_LIBRARY-NOTFOUND |
| CUDA_nppicom_static_LIBRARY | CUDA_nppicom_static_LIBRARY-NOTFOUND |
| CUDA_nppidei_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppidei.lib |
| CUDA_nppidei_static_LIBRARY | CUDA_nppidei_static_LIBRARY-NOTFOUND |
| CUDA_nppif_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppif.lib |
| CUDA_nppif_static_LIBRARY | CUDA_nppif_static_LIBRARY-NOTFOUND |
| CUDA_nppig_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppig.lib |
| CUDA_nppig_static_LIBRARY | CUDA_nppig_static_LIBRARY-NOTFOUND |
| CUDA_nppim_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppim.lib |
| CUDA_nppim_static_LIBRARY | CUDA_nppim_static_LIBRARY-NOTFOUND |
| CUDA_nppist_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppist.lib |
| CUDA_nppist_static_LIBRARY | CUDA_nppist_static_LIBRARY-NOTFOUND |
| CUDA_nppisu_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppisu.lib |
| CUDA_nppisu_static_LIBRARY | CUDA_nppisu_static_LIBRARY-NOTFOUND |
| CUDA_nppitc_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppitc.lib |
| CUDA_nppitc_static_LIBRARY | CUDA_nppitc_static_LIBRARY-NOTFOUND |
| CUDA_npps_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/npps.lib |
| CUDA_npps_static_LIBRARY | CUDA_npps_static_LIBRARY-NOTFOUND |
| CUDA_nvToolsExt_LIBRARY | C:/Program Files/NVIDIA Corporation/NvToolsExt/lib/x64/nvToolsExt64_1.lib |
| CUDA_nvcuvenc_LIBRARY | CUDA_nvcuvenc_LIBRARY-NOTFOUND |
| CUDA_nvcuvid_LIBRARY | CUDA_nvcuvid_LIBRARY-NOTFOUND |
| CUDA_nvgraph_LIBRARY | CUDA_nvgraph_LIBRARY-NOTFOUND |
| CUDA_nvgraph_static_LIBRARY | CUDA_nvgraph_static_LIBRARY-NOTFOUND |



| Name | Value |
|---|---|
| CUDA_nppc_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppc.lib |
| CUDA_nppc_static_LIBRARY | CUDA_nppc_static_LIBRARY-NOTFOUND |
| CUDA_nppi_LIBRARY | CUDA_nppi_LIBRARY-NOTFOUND |
| CUDA_nppial_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppial.lib |
| CUDA_nppial_static_LIBRARY | CUDA_nppial_static_LIBRARY-NOTFOUND |
| CUDA_nppicc_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppicc.lib |
| CUDA_nppicc_static_LIBRARY | CUDA_nppicc_static_LIBRARY-NOTFOUND |
| CUDA_nppicom_LIBRARY | CUDA_nppicom_LIBRARY-NOTFOUND |
| CUDA_nppicom_static_LIBRARY | CUDA_nppicom_static_LIBRARY-NOTFOUND |
| CUDA_nppidei_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppidei.lib |
| CUDA_nppidei_static_LIBRARY | CUDA_nppidei_static_LIBRARY-NOTFOUND |
| CUDA_nppif_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppif.lib |
| CUDA_nppif_static_LIBRARY | CUDA_nppif_static_LIBRARY-NOTFOUND |
| CUDA_nppig_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppig.lib |
| CUDA_nppig_static_LIBRARY | CUDA_nppig_static_LIBRARY-NOTFOUND |
| CUDA_nppim_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppim.lib |
| CUDA_nppim_static_LIBRARY | CUDA_nppim_static_LIBRARY-NOTFOUND |
| CUDA_nppist_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppist.lib |
| CUDA_nppist_static_LIBRARY | CUDA_nppist_static_LIBRARY-NOTFOUND |
| CUDA_nppisu_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppisu.lib |
| CUDA_nppisu_static_LIBRARY | CUDA_nppisu_static_LIBRARY-NOTFOUND |
| CUDA_nppitc_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nppitc.lib |
| CUDA_nppitc_static_LIBRARY | CUDA_nppitc_static_LIBRARY-NOTFOUND |
| CUDA_npps_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/npps.lib |
| CUDA_npps_static_LIBRARY | CUDA_npps_static_LIBRARY-NOTFOUND |
| CUDA_nvToolsExt_LIBRARY | C:/Program Files/NVIDIA Corporation/NvToolsExt/lib/x64/nvToolsExt64_1.lib |
| CUDA_nvcuvenc_LIBRARY | CUDA_nvcuvenc_LIBRARY-NOTFOUND |
| CUDA_nvcuvid_LIBRARY | CUDA_nvcuvid_LIBRARY-NOTFOUND |
| CUDA_nvgraph_LIBRARY | CUDA_nvgraph_LIBRARY-NOTFOUND |
| CUDA_nvgraph_static_LIBRARY | CUDA_nvgraph_static_LIBRARY-NOTFOUND |
| CUDA_nvjpeg_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nvjpeg.lib |
| CUDA_nvjpeg_static_LIBRARY | CUDA_nvjpeg_static_LIBRARY-NOTFOUND |
| CUDA_nvml_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nvml.lib |
| CUDA_nvrtc_LIBRARY | C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/v11.8/lib/x64/nvrtc.lib |
| Caffe2_DIR | C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/share/cmake/Caffe2 |
| MKLDNN_DIR | MKLDNN_DIR-NOTFOUND |
| MKL_DIR | MKL_DIR-NOTFOUND |
| NP_CAMERASDK_LIBRARY | C:/Program Files (x86)/OptiTrack/Camera SDK/lib/CameraLibrary2015x64D.lib |
| OpenCV_DIR | C:/mahdi/VPTV_VS2022/packages/opencv-4.8.0/build |
| Qt5Core_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Core |
| Qt5Gui_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Gui |
| Qt5Sql_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Sql |
| Qt5Widgets_DIR | C:/Qt/Qt5.9.9/5.9.9/msvc2017_64/lib/cmake/Qt5Widgets |
| TORCH_LIBRARY | C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/lib/torch.lib |
| Torch_DIR | C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/share/cmake/Torch |
| VTK_DIR | C:/Program Files/VTK/lib/cmake/vtk-8.2 |
| YAMLCPP_HAVE_H | TRUE |
| YAMLCPP_H_INCLUDE_DIR | C:/mahdi/VPTV_VS2022/packages/YAML_CPP0.6/YAML_CPP 0.6/include |
| YAMLCPP_LIBRARY | C:/mahdi/VPTV_VS2022/packages/YAML_CPP0.6/YAML_CPP 0.6/lib/yaml-cpp.lib |
| c10_LIBRARY | C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/lib/c10.lib |
| kineto_LIBRARY | C:/mahdi/VPTV_VS2022/packages/libtorch-win-shared-with-deps-2.1.0+cu118/libtorch/lib/kineto.lib |



Solution 'LPT' (17 of 17 projects)
- ALL_BUILD
- app-LPT_Optitrack
- app-LPT_Virtual
- module-calib
- module-core
- module-correspond
- module-correspondcuda
- module-dataaquisition
- module-datagen
- module-imageproc
- module-tracking
- module-visualization
- test-mp1_synchronization
- test-mp3_imageProcessing
- test-mp4_correspondence
- test-mp5_tracking
- ZERO_CHECK



```
12>test-mp4_correspondence.vcxproj -> C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo\test-mp4_correspondence.exe
12>Done building project "test-mp4_correspondence.vcxproj".
13>test-mp3_imageProcessing.vcxproj -> C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo\test-mp3_imageProcessing.exe
15>app-LPT_Virtual.vcxproj -> C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo\app-LPT_Virtual.exe
16>app-LPT_Optitrack.vcxproj -> C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo\app-LPT_Optitrack.exe
15>Done building project "app-LPT_Virtual.vcxproj".
13>Done building project "test-mp3_imageProcessing.vcxproj".
16>Done building project "app-LPT_Optitrack.vcxproj".
14>test-mp1_synchronization.vcxproj -> C:\mahdi\VPTV_VS2022\build\bin\RelWithDebInfo\test-mp1_synchronization.exe
14>Done building project "test-mp1_synchronization.vcxproj".
17>------ Rebuild All started: Project: ALL_BUILD, Configuration: RelWithDebInfo x64 ------
17>Building Custom Rule C:/mahdi/VPTV_VS2022/CMakeLists.txt
========== Rebuild All: 17 succeeded, 0 failed, 0 skipped ==========
========== Rebuild started at 11:27 AM and took 37.315 seconds ==========
```