Laboratorium zamykające – (czas: 2 laboratoria)
Podstawy kompilacji bibliotek i oprogramowania za pomocą CMAKE na przykładzie programu wykorzystującego OpenCV.

Miłosz Martynow
Laboratoria z Administracji i Architektury Systemów Operacyjnych
Wydział Fizyki Technicznej i Matematyki Stosowanej
Zakład Fizyki Teoretycznej i Informatyki Kwantowej
Zajęcia zdalne ze względu na kwarantanne ze względu na COVID-19 na Politechnice Gdańskiej

1. Instalacja i przetestowanie OpenCV dla C++/Python (1 lab)
2. Napisanie własnego programu do analizy obrazu (1 lab)

Wstęp

Na zeszłym laboratorium zapoznaliśmy się z kompilacją oprogramowania za pomocą *make*. Jest to rozwiązanie bardzo wygodnie dla wielu projektów. Niemniej jednak możemy się spotkać z wieloma bardzo użytecznymi programami i bibliotekami, które nie posiadają pliku make – tak też ich kompilacja może stanowić problem ze względu na dużą ilość kodu zewnętrznego wykorzystywanego przez dane rozwiązanie.

Instalacja OpenCV

Biblioteka OpenCV jest olbrzymią i darmową biblioteką, którą można wykorzystać do bardzo wydajnej obróbki obrazu – w tym zdjęć, filmów, streamingów itd. Z biblioteki OpenCV można łatwo korzystać za pomocą języka C++ i Python, ale także istnieją rozwiązania dla innych środowisk. OpenCV zawiera bardzo dużo modułów opcjonalnych takich jak na przykład moduł DNN, który pozwala na prace z wytrenowanymi modelami uczenia maszynowego i sieci neuronowych ogólnie. Wiele platform jak Raspberry Pi czy NVIDIA Jetson korzystają z wbudowanych systemów operacyjnych Linux i umożliwiają łatwą komunikacje z takimi urządzeniami peryferyjnymi jak kamery – tym samym umożliwiają obróbkę obrazu z łatwym pisaniem programów testowych i końcowych z wykorzystaniem np. OpenCV już jako produkt.

Tak jak w poprzednim projekcie zaznaczam, że nie mam możliwości przetestowania instalacji na wielu komputerach – sama COVIDowa sytuacja sprawia, że nie działamy na „laboratoryjnych" komputerach, gdzie wszystko jest znormalizowane albo chociaż znane. Ale z drugiej strony to dobrze – jeśli coś nie zadziała, nie zainstaluje się itp. to przy rozwiązywaniu tego typu problemów najwięcej się uczymy – mówię to całkowicie poważnie i bez sarkazmu. Takie sytuacje w pracy, projektach itd. będą na każdym kroku i prawdziwego kodowania ostanie się może 10% waszego czasu prac :) Poniżej zamieszczam dwa linki do opcjonalnej instalacji, gdzie link [1] jest genialnym opisem wykorzystywanych bibliotek itp – pozwoliłem sobie nawet nie opisywać wielu instalowanych zależności itp. bo tam to jest pięknie wyłożone.

[1] https://cv-tricks.com/installation/opencv-4-1-ubuntu18-04/
[2] https://www.learnopencv.com/install-opencv-4-on-ubuntu-18-04/

1. Instalacja pakietów systemowych – DEPENDENCIES

```
# Aktualizacja
sudo apt-get update
sudo apt-get upgrade

# Folder do pracy
mkdir AAOS_proj3
cd AAOS_proj3
pwd <ŚĆEŻKA CWD - CURRENT WORKING DIRECTORY>

# PODSTAWOWE BIBLIOTEKI – OpenCV będzie w stanie odnaleźć ich domyślne lokalizacje
sudo apt-get install build-essential \
                     checkinstall \
                     pkg-config \
                     yasm \
                     cmake \
                     git \
                     gfortran \
                     libeigen3-dev \
                     libblas-dev \
                     liblapacke-dev \
                     libprotobuf-dev \
                     protobuf-compiler \
                     libgflags-dev \
                     libjpeg8-dev \
                     libpng-dev \
                     libtiff-dev

# PROBLEMY Z LIBJASPER - wymagany pakiet przy niektórych modułach
sudo apt-get install software-properties-common
sudo add-apt-repository "deb http://security.ubuntu.com/ubuntu xenial-security main"
sudo apt-get update
sudo apt-get install libjasper1

# INSTALACJA KODEKÓW - wraz z dodaniem odpowiednich headerów do systemy
sudo apt-get install libavcodec-dev \
                     libavformat-dev \
                     libswscale-dev \
                     libdc1394-22-dev \
                     libxine2-dev \
                     libv4l-dev
cd /usr/include/linux
sudo ln -s -f ../libv4l1-videodev.h videodev.h
cd <ŚĆEŻKA CWD - CURRENT WORKING DIRECTORY>

# BIBLIOTEKI DO MIĘDZY INNYMI DO STREAMINGU
sudo apt-get install libgstreamer1.0-dev \
                     libgstreamer-plugins-base1.0-dev \
                     libgtk2.0-dev \
                     libtbb-dev \
                     qt5-default \
                     libatlas-base-dev \
                     libfaac-dev \
                     libmp3lame-dev \
                     libtheora-dev \
                     libvorbis-dev \
                     libxvidcore-dev \
                     libopencore-amrnb-dev \
                     libopencore-amrwb-dev \
                     libavresample-dev \
                     x264 \
                     v4l-utils
```

2. Pobranie OpenCV wraz z dodatkowymi modułami

```
cd <ŚĆEŻKA CWD - CURRENT WORKING DIRECTORY>
git clone https://github.com/opencv/opencv.git # Klonowanie repozytorium OpenCV
cd opencv
git checkout master # Dla pewności, że jesteśmy na master branch
cd ../

git clone https://github.com/opencv/opencv_contrib.git
cd opencv_contrib
git checkout master
cd ../
```

3. Kompilacja – zazwyczaj przy korzystaniu z CMAKE, warto stworzyć w folderze oprogramowania kompilowanego folder *build*, z którego będziemy wywoływać instalacje itd. Wywołany program CMAKE. Ustawiając flagi -D możemy sterować procesem przygotowywania instalacji biblioteki – coś w stylu Windowsowej instalacji „zaawansowane ustawienia instalacji” gdzie wybieramy co dokładnie chcemy zainstalować itp. wiele można wywnioskować już z samych nazw, np. WITH_PYTHON=ON przygotuje kompilacje także dla Pythona – jeśli ten jest zainstalowany w folderze domyślnym CMAKE, sam go odnajdzie, jeśli jednak kompilować chcemy dla konkretnej wersji, to musimy podać ścieżkę we fladze OPENCV_PYTHON3_INSTALL_PATH=<path to python3>. Proces instalacji wygląda następująco: wpierw tworzymy folder *build*, w którym będziemy wywoływali program *cmake* z odpowiednimi flagami, on nam przygotuje środowisko i plik *makefile*, który umożliwi wywołanie polecenia *make*, które uruchomi naszą kompilacje/instalacje

```
cd opencv
mkdir build
cd build

cmake -D CMAKE_BUILD_TYPE=RELEASE \
       -D CMAKE_INSTALL_PREFIX=/usr/local/ \
       -D INSTALL_C_EXAMPLES=OFF \
       -D INSTALL_PYTHON_EXAMPLES=OFF \
       -D WITH_TBB=ON \
       -D WITH_V4L=ON \
       -D WITH_QT=ON \
       -D WITH_OPENGL=ON \
       -D WITH_PYTHON=ON \
       -D OPENCV_EXTRA_MODULES_PATH=../../opencv_contrib/modules \
       -D BUILD_EXAMPLES=OFF \
       ../
```

Przejrzyj dokładnie to co wypisał nasz CMAKE – zauważysz, że zbudowały się zrównoleglające frameworki jak TBB (Parallel framework: TBB), Odnaleziona została ścieżka do Pythona czy do pakietów optymalizujących i przyspieszających matmę (i ją zęsto umożliwiającą) w postaci pakietów LAPACK, EIGEN w sekcji Other third-party libraries. Swój output z CMAKE zamieszczam na końcu, można go porównać. Generalnie jego analiza zaczyna być bardzo ważna jeśli chcą Państwo zbudować OpenCV dla kart graficznych NVIDII/GPU itd. To on na mówi, czy jest po co kompilować czy jeszcze czegoś nie dołączyliśmy itp

4. Jak już zbudowaliśmy środowisko, teraz należy wywołać kompilacje. Może ona potrwać chwilę, więc warto ją zrównoleglić na kilka procesorów za pomocą -jN, gdzie N to ilość procesorów:
   ```
   make -j8
   sudo make install
   sudo ldconfig
   ```
5. Aby przetestować, poprawność instalacji biblioteki OpenCV wywołajmy podstawowy program w C++, który wyświetli obraz Lenny Na przykładzie kodu ze strony producenta. https://docs.opencv.org/2.4/doc/tutorials/introduction/linux_gcc_cmake/linux_gcc_cmake.html
   ```
   mkdir test_opencv
   cd test_opencv
   vi test_opencv.cpp
   ```

```
#include <stdio.h>
#include <opencv2/opencv.hpp>

using namespace cv;

int main(int argc, char** argv )
{
    if ( argc != 2 )
    {
        printf("usage: DisplayImage.out <Image_Path>\n");
        return -1;
    }

    Mat image;
    image = imread( argv[1], 1 );

    if ( !image.data )
    {
        printf("No image data \n");
        return -1;
    }
    namedWindow("Display Image", WINDOW_AUTOSIZE );
    imshow("Display Image", image);

    waitKey(0);

    return 0;
}
```

Taki program skompilujemy za pomocą własnego podstawowego CMAKE. To czego poszukuje program CMAKE, to plik CMakeLists.txt, w którym zamieszczone są wszystkie informacje dotyczące kompilacje, jak adres folderu z libs (-L), include (-I) itd. W związku z tym, że dokonaliśmy instalacji w domyślnym folderze *usr/local/* możemy napisać bardzo krótki plik CMakeLists.txt i wywołać program cmake . , gdzie kropka oznacza wskazanie foldeu gdzie cmake powinien szukać CMakeList.txt – dla tego przy konfiguracji OpenCV na końcu CMAKE były .., bo CMakeList.txt był folder niżej od utworzonego folderu build.

```
vi CmakeList.cpp

cmake_minimum_required(VERSION 2.8)
project( test_opencv_cpp )
find_package( OpenCV REQUIRED )
add_executable( test_opencv_cpp test_opencv_cpp.cpp )
target_link_libraries( test_opencv_cpp ${OpenCV_LIBS} )

cmake . # konfiguracja kompilacji
make # kompilacja
./test_opencv_cpp ../Lenna.png # ja trzymam zdjęcie Lenny folder niżej
```

Pisanie plików CmakeList.txt, to bardzo duży temat – któremu należało by poświęcić parę laboratoriów. Dla osób chcących się rozwijać w technologiach CPP, jest to narzędzie *must-have* i zachęcam do samodzielnego zgłębiania tego tematu.

6. Napisz program w C++ wykorzystujący OpenCV, kompilowany za pomocą CMAKE, który odnajdzie i zaznaczy twarz – np. na obrazie Lenny, albo streamie z kamerki laptopa. Możesz wykorzystać Haar Feature Cascade Classifier albo modele DNN (przykład https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html). Napisz raport w którym opisałeś co i w jaki sposób zrobiłeś – dokładność punktacji 1p, można uzyskać max 5p. Ze względu na dowolność projektu, ilość punktów tym razem jest uzależniona tylko od jakości raportu – i od jakości opisu procedur – kody zamieszczaj w teście. Raport wyślij w formie PDF. Dead line 12.06.2020 włącznie.
7. Output z CMAKE:

```
-- The CXX compiler identification is GNU 9.3.0
-- The C compiler identification is GNU 9.3.0
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Detected processor: x86_64
-- Found PythonInterp: /usr/bin/python2.7 (found suitable version "2.7.18", minimum required is "2.7")
-- Could NOT find PythonLibs (missing: PYTHON_LIBRARIES PYTHON_INCLUDE_DIRS) (Required is exact version "2.7.18")
Traceback (most recent call last):
  File "<string>", line 1, in <module>
ImportError: No module named numpy.distutils
-- Found PythonInterp: /usr/bin/python3 (found suitable version "3.8.2", minimum required is "3.2")
-- Found PythonLibs: /usr/lib/x86_64-linux-gnu/libpython3.8.so (found suitable exact version "3.8.2")
-- Looking for ccache - not found
-- Performing Test HAVE_CXX_FSIGNED_CHAR
-- Performing Test HAVE_CXX_FSIGNED_CHAR - Success
-- Performing Test HAVE_C_FSIGNED_CHAR
-- Performing Test HAVE_C_FSIGNED_CHAR - Success
-- Performing Test HAVE_CXX_W
-- Performing Test HAVE_CXX_W - Success
-- Performing Test HAVE_C_W
-- Performing Test HAVE_C_W - Success
-- Performing Test HAVE_CXX_WALL
-- Performing Test HAVE_CXX_WALL - Success
-- Performing Test HAVE_C_WALL
-- Performing Test HAVE_C_WALL - Success
-- Performing Test HAVE_CXX_WERROR_RETURN_TYPE
-- Performing Test HAVE_CXX_WERROR_RETURN_TYPE - Success
-- Performing Test HAVE_C_WERROR_RETURN_TYPE
-- Performing Test HAVE_C_WERROR_RETURN_TYPE - Success
-- Performing Test HAVE_CXX_WERROR_NON_VIRTUAL_DTOR
-- Performing Test HAVE_CXX_WERROR_NON_VIRTUAL_DTOR - Success
-- Performing Test HAVE_C_WERROR_NON_VIRTUAL_DTOR
-- Performing Test HAVE_C_WERROR_NON_VIRTUAL_DTOR - Failed
-- Performing Test HAVE_CXX_WERROR_ADDRESS
-- Performing Test HAVE_CXX_WERROR_ADDRESS - Success
-- Performing Test HAVE_C_WERROR_ADDRESS
-- Performing Test HAVE_C_WERROR_ADDRESS - Success
-- Performing Test HAVE_CXX_WERROR_SEQUENCE_POINT
-- Performing Test HAVE_CXX_WERROR_SEQUENCE_POINT - Success
-- Performing Test HAVE_C_WERROR_SEQUENCE_POINT
-- Performing Test HAVE_C_WERROR_SEQUENCE_POINT - Success
-- Performing Test HAVE_CXX_WFORMAT
-- Performing Test HAVE_CXX_WFORMAT - Success
-- Performing Test HAVE_C_WFORMAT
-- Performing Test HAVE_C_WFORMAT - Success
-- Performing Test HAVE_CXX_WERROR_FORMAT_SECURITY
-- Performing Test HAVE_CXX_WERROR_FORMAT_SECURITY - Success
-- Performing Test HAVE_C_WERROR_FORMAT_SECURITY
-- Performing Test HAVE_C_WERROR_FORMAT_SECURITY - Success
-- Performing Test HAVE_CXX_WMISSING_DECLARATIONS
-- Performing Test HAVE_CXX_WMISSING_DECLARATIONS - Success
-- Performing Test HAVE_C_WMISSING_DECLARATIONS
-- Performing Test HAVE_C_WMISSING_DECLARATIONS - Success
-- Performing Test HAVE_CXX_WMISSING_PROTOTYPES
-- Performing Test HAVE_CXX_WMISSING_PROTOTYPES - Failed
-- Performing Test HAVE_C_WMISSING_PROTOTYPES
-- Performing Test HAVE_C_WMISSING_PROTOTYPES - Success
-- Performing Test HAVE_CXX_WSTRICT_PROTOTYPES
-- Performing Test HAVE_CXX_WSTRICT_PROTOTYPES - Failed
-- Performing Test HAVE_C_WSTRICT_PROTOTYPES
-- Performing Test HAVE_C_WSTRICT_PROTOTYPES - Success
-- Performing Test HAVE_CXX_WUNDEF
-- Performing Test HAVE_CXX_WUNDEF - Success
-- Performing Test HAVE_C_WUNDEF
-- Performing Test HAVE_C_WUNDEF - Success
-- Performing Test HAVE_CXX_WINIT_SELF
-- Performing Test HAVE_CXX_WINIT_SELF - Success
-- Performing Test HAVE_C_WINIT_SELF
-- Performing Test HAVE_C_WINIT_SELF - Success
-- Performing Test HAVE_CXX_WPOINTER_ARITH
-- Performing Test HAVE_CXX_WPOINTER_ARITH - Success
-- Performing Test HAVE_C_WPOINTER_ARITH
-- Performing Test HAVE_C_WPOINTER_ARITH - Success
-- Performing Test HAVE_CXX_WSHADOW
-- Performing Test HAVE_CXX_WSHADOW - Success
-- Performing Test HAVE_C_WSHADOW
-- Performing Test HAVE_C_WSHADOW - Success
-- Performing Test HAVE_CXX_WSIGN_PROMO
-- Performing Test HAVE_CXX_WSIGN_PROMO - Success
-- Performing Test HAVE_C_WSIGN_PROMO
-- Performing Test HAVE_C_WSIGN_PROMO - Failed
-- Performing Test HAVE_CXX_WUNINITIALIZED
-- Performing Test HAVE_CXX_WUNINITIALIZED - Success
-- Performing Test HAVE_C_WUNINITIALIZED
-- Performing Test HAVE_C_WUNINITIALIZED - Success
-- Performing Test HAVE_CXX_WSUGGEST_OVERRIDE
-- Performing Test HAVE_CXX_WSUGGEST_OVERRIDE - Success
-- Performing Test HAVE_C_WSUGGEST_OVERRIDE
-- Performing Test HAVE_C_WSUGGEST_OVERRIDE - Failed
-- Performing Test HAVE_CXX_WNO_DELETE_NON_VIRTUAL_DTOR
-- Performing Test HAVE_CXX_WNO_DELETE_NON_VIRTUAL_DTOR - Success
-- Performing Test HAVE_C_WNO_DELETE_NON_VIRTUAL_DTOR
-- Performing Test HAVE_C_WNO_DELETE_NON_VIRTUAL_DTOR - Failed
-- Performing Test HAVE_CXX_WNO_UNNAMED_TYPE_TEMPLATE_ARGS
-- Performing Test HAVE_CXX_WNO_UNNAMED_TYPE_TEMPLATE_ARGS - Failed
-- Performing Test HAVE_C_WNO_UNNAMED_TYPE_TEMPLATE_ARGS
-- Performing Test HAVE_C_WNO_UNNAMED_TYPE_TEMPLATE_ARGS - Failed
-- Performing Test HAVE_CXX_WNO_COMMENT
-- Performing Test HAVE_CXX_WNO_COMMENT - Success
-- Performing Test HAVE_C_WNO_COMMENT
-- Performing Test HAVE_C_WNO_COMMENT - Success
-- Performing Test HAVE_CXX_WIMPLICIT_FALLTHROUGH_3
-- Performing Test HAVE_CXX_WIMPLICIT_FALLTHROUGH_3 - Success
-- Performing Test HAVE_C_WIMPLICIT_FALLTHROUGH_3
-- Performing Test HAVE_C_WIMPLICIT_FALLTHROUGH_3 - Success
-- Performing Test HAVE_CXX_WNO_STRICT_OVERFLOW
-- Performing Test HAVE_CXX_WNO_STRICT_OVERFLOW - Success
-- Performing Test HAVE_C_WNO_STRICT_OVERFLOW
-- Performing Test HAVE_C_WNO_STRICT_OVERFLOW - Success
-- Performing Test HAVE_CXX_FDIAGNOSTICS_SHOW_OPTION
-- Performing Test HAVE_CXX_FDIAGNOSTICS_SHOW_OPTION - Success
-- Performing Test HAVE_C_FDIAGNOSTICS_SHOW_OPTION
-- Performing Test HAVE_C_FDIAGNOSTICS_SHOW_OPTION - Success
-- Performing Test HAVE_CXX_WNO_LONG_LONG
-- Performing Test HAVE_CXX_WNO_LONG_LONG - Success
-- Performing Test HAVE_C_WNO_LONG_LONG
-- Performing Test HAVE_C_WNO_LONG_LONG - Success
-- Performing Test HAVE_CXX_PTHREAD
-- Performing Test HAVE_CXX_PTHREAD - Success
-- Performing Test HAVE_C_PTHREAD
-- Performing Test HAVE_C_PTHREAD - Success
-- Performing Test HAVE_CXX_FOMIT_FRAME_POINTER
-- Performing Test HAVE_CXX_FOMIT_FRAME_POINTER - Success
-- Performing Test HAVE_C_FOMIT_FRAME_POINTER
-- Performing Test HAVE_C_FOMIT_FRAME_POINTER - Success
-- Performing Test HAVE_CXX_FFUNCTION_SECTIONS
-- Performing Test HAVE_CXX_FFUNCTION_SECTIONS - Success
-- Performing Test HAVE_C_FFUNCTION_SECTIONS
-- Performing Test HAVE_C_FFUNCTION_SECTIONS - Success
-- Performing Test HAVE_CXX_FDATA_SECTIONS
-- Performing Test HAVE_CXX_FDATA_SECTIONS - Success
-- Performing Test HAVE_C_FDATA_SECTIONS
-- Performing Test HAVE_C_FDATA_SECTIONS - Success
-- Performing Test HAVE_CXX_MSSE (check file: cmake/checks/cpu_sse.cpp)
-- Performing Test HAVE_CXX_MSSE - Success
-- Performing Test HAVE_CXX_MSSE2 (check file: cmake/checks/cpu_sse2.cpp)
-- Performing Test HAVE_CXX_MSSE2 - Success
-- Performing Test HAVE_CXX_MSSE3 (check file: cmake/checks/cpu_sse3.cpp)
-- Performing Test HAVE_CXX_MSSE3 - Success
-- Performing Test HAVE_CXX_MSSSE3 (check file: cmake/checks/cpu_ssse3.cpp)
-- Performing Test HAVE_CXX_MSSSE3 - Success
-- Performing Test HAVE_CXX_MSSE4_1 (check file: cmake/checks/cpu_sse41.cpp)
-- Performing Test HAVE_CXX_MSSE4_1 - Success
-- Performing Test HAVE_CXX_MPOPCNT (check file: cmake/checks/cpu_popcnt.cpp)
-- Performing Test HAVE_CXX_MPOPCNT - Success
-- Performing Test HAVE_CXX_MSSE4_2 (check file: cmake/checks/cpu_sse42.cpp)
-- Performing Test HAVE_CXX_MSSE4_2 - Success
-- Performing Test HAVE_CXX_MF16C (check file: cmake/checks/cpu_fp16.cpp)
-- Performing Test HAVE_CXX_MF16C - Success
-- Performing Test HAVE_CXX_MFMA
-- Performing Test HAVE_CXX_MFMA - Success
-- Performing Test HAVE_CXX_MAVX (check file: cmake/checks/cpu_avx.cpp)
-- Performing Test HAVE_CXX_MAVX - Success
-- Performing Test HAVE_CXX_MAVX2 (check file: cmake/checks/cpu_avx2.cpp)
-- Performing Test HAVE_CXX_MAVX2 - Success
-- Performing Test HAVE_CXX_MAVX512F (check file: cmake/checks/cpu_avx512.cpp)
-- Performing Test HAVE_CXX_MAVX512F - Success
-- Performing Test HAVE_CXX_MAVX512F_MAVX512CD (check file: cmake/checks/cpu_avx512common.cpp)
-- Performing Test HAVE_CXX_MAVX512F_MAVX512CD - Success
-- Performing Test HAVE_CXX_MAVX512F_MAVX512CD_MAVX512VL_MAVX512BW_MAVX512DQ (check file: cmake/checks/cpu_avx512skx.cpp)
-- Performing Test HAVE_CXX_MAVX512F_MAVX512CD_MAVX512VL_MAVX512BW_MAVX512DQ - Success
-- Performing Test HAVE_CPU_BASELINE_FLAGS
-- Performing Test HAVE_CPU_BASELINE_FLAGS - Success
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_SSE4_1
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_SSE4_1 - Success
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_SSE4_2
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_SSE4_2 - Success
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_FP16
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_FP16 - Success
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_AVX
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_AVX - Success
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_AVX2
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_AVX2 - Success
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_AVX512_SKX
-- Performing Test HAVE_CPU_DISPATCH_FLAGS_AVX512_SKX - Success
-- Performing Test HAVE_CXX_FVISIBILITY_HIDDEN
-- Performing Test HAVE_CXX_FVISIBILITY_HIDDEN - Success
-- Performing Test HAVE_C_FVISIBILITY_HIDDEN
-- Performing Test HAVE_C_FVISIBILITY_HIDDEN - Success
-- Performing Test HAVE_CXX_FVISIBILITY_INLINES_HIDDEN
-- Performing Test HAVE_CXX_FVISIBILITY_INLINES_HIDDEN - Success
-- Performing Test HAVE_C_FVISIBILITY_INLINES_HIDDEN
-- Performing Test HAVE_C_FVISIBILITY_INLINES_HIDDEN - Failed
-- Performing Test HAVE_LINK_AS_NEEDED
-- Performing Test HAVE_LINK_AS_NEEDED - Success
-- Looking for pthread.h
-- Looking for pthread.h - found
-- Looking for posix_memalign
-- Looking for posix_memalign - found
-- Looking for malloc.h
-- Looking for malloc.h - found
-- Looking for memalign
-- Looking for memalign - found
-- Check if the system is big endian
-- Searching 16 bit integer
-- Looking for sys/types.h
-- Looking for sys/types.h - found
-- Looking for stdint.h
-- Looking for stdint.h - found
-- Looking for stddef.h
-- Looking for stddef.h - found
-- Check size of unsigned short
-- Check size of unsigned short - done
-- Using unsigned short
-- Check if the system is big endian - little endian
-- Found ZLIB: /usr/lib/x86_64-linux-gnu/libz.so (found suitable version "1.2.11", minimum required is "1.2.3")
-- Found JPEG: /usr/lib/x86_64-linux-gnu/libjpeg.so (found version "80")
-- Found TIFF: /usr/lib/x86_64-linux-gnu/libtiff.so (found version "4.1.0")
-- Performing Test HAVE_C_STD_C99
-- Performing Test HAVE_C_STD_C99 - Success
-- Performing Test HAVE_C_WNO_UNUSED_VARIABLE
-- Performing Test HAVE_C_WNO_UNUSED_VARIABLE - Success
-- Performing Test HAVE_C_WNO_UNUSED_FUNCTION
-- Performing Test HAVE_C_WNO_UNUSED_FUNCTION - Success
-- Performing Test HAVE_C_WNO_SHADOW
-- Performing Test HAVE_C_WNO_SHADOW - Success
-- Performing Test HAVE_C_WNO_MAYBE_UNINITIALIZED
-- Performing Test HAVE_C_WNO_MAYBE_UNINITIALIZED - Success
-- Performing Test HAVE_C_WNO_MISSING_PROTOTYPES
-- Performing Test HAVE_C_WNO_MISSING_PROTOTYPES - Success
-- Performing Test HAVE_C_WNO_MISSING_DECLARATIONS
-- Performing Test HAVE_C_WNO_MISSING_DECLARATIONS - Success
-- Performing Test HAVE_C_WNO_IMPLICIT_FALLTHROUGH
-- Performing Test HAVE_C_WNO_IMPLICIT_FALLTHROUGH - Success
-- Could NOT find OpenJPEG (minimal suitable version: 2.0, recommended version >= 2.3.1)
-- Could NOT find Jasper (missing: JASPER_LIBRARIES JASPER_INCLUDE_DIR)
-- Performing Test HAVE_C_WNO_IMPLICIT_FUNCTION_DECLARATION
-- Performing Test HAVE_C_WNO_IMPLICIT_FUNCTION_DECLARATION - Success
-- Performing Test HAVE_C_WNO_UNINITIALIZED
-- Performing Test HAVE_C_WNO_UNINITIALIZED - Success
-- Performing Test HAVE_C_WNO_UNUSED_BUT_SET_PARAMETER
-- Performing Test HAVE_C_WNO_UNUSED_BUT_SET_PARAMETER - Success
-- Performing Test HAVE_C_WNO_UNUSED
-- Performing Test HAVE_C_WNO_UNUSED - Success
-- Performing Test HAVE_C_WNO_SIGN_COMPARE
-- Performing Test HAVE_C_WNO_SIGN_COMPARE - Success
-- Performing Test HAVE_C_WNO_POINTER_COMPARE
-- Performing Test HAVE_C_WNO_POINTER_COMPARE - Success
-- Performing Test HAVE_C_WNO_ABSOLUTE_VALUE
-- Performing Test HAVE_C_WNO_ABSOLUTE_VALUE - Success
-- Performing Test HAVE_C_WNO_UNUSED_PARAMETER
-- Performing Test HAVE_C_WNO_UNUSED_PARAMETER - Success
-- Performing Test HAVE_C_WNO_STRICT_PROTOTYPES
-- Performing Test HAVE_C_WNO_STRICT_PROTOTYPES - Success
-- Found ZLIB: /usr/lib/x86_64-linux-gnu/libz.so (found version "1.2.11")
-- Found PNG: /usr/lib/x86_64-linux-gnu/libpng.so (found version "1.6.37")
-- Looking for /usr/include/libpng/png.h
-- Looking for /usr/include/libpng/png.h - found
-- Looking for semaphore.h
-- Looking for semaphore.h - found
-- Performing Test HAVE_CXX_WNO_SHADOW
-- Performing Test HAVE_CXX_WNO_SHADOW - Success
-- Performing Test HAVE_CXX_WNO_UNUSED
-- Performing Test HAVE_CXX_WNO_UNUSED - Success
-- Performing Test HAVE_CXX_WNO_SIGN_COMPARE
-- Performing Test HAVE_CXX_WNO_SIGN_COMPARE - Success
-- Performing Test HAVE_CXX_WNO_UNDEF
-- Performing Test HAVE_CXX_WNO_UNDEF - Success
-- Performing Test HAVE_CXX_WNO_MISSING_DECLARATIONS
-- Performing Test HAVE_CXX_WNO_MISSING_DECLARATIONS - Success
-- Performing Test HAVE_CXX_WNO_UNINITIALIZED
-- Performing Test HAVE_CXX_WNO_UNINITIALIZED - Success
-- Performing Test HAVE_CXX_WNO_SWITCH
-- Performing Test HAVE_CXX_WNO_SWITCH - Success
-- Performing Test HAVE_CXX_WNO_PARENTHESES
-- Performing Test HAVE_CXX_WNO_PARENTHESES - Success
-- Performing Test HAVE_CXX_WNO_ARRAY_BOUNDS
-- Performing Test HAVE_CXX_WNO_ARRAY_BOUNDS - Success
-- Performing Test HAVE_CXX_WNO_EXTRA
-- Performing Test HAVE_CXX_WNO_EXTRA - Success
-- Performing Test HAVE_CXX_WNO_DEPRECATED_DECLARATIONS
-- Performing Test HAVE_CXX_WNO_DEPRECATED_DECLARATIONS - Success
-- Performing Test HAVE_CXX_WNO_MISLEADING_INDENTATION
-- Performing Test HAVE_CXX_WNO_MISLEADING_INDENTATION - Success
-- Performing Test HAVE_CXX_WNO_DEPRECATED
-- Performing Test HAVE_CXX_WNO_DEPRECATED - Success
-- Performing Test HAVE_CXX_WNO_SUGGEST_OVERRIDE
-- Performing Test HAVE_CXX_WNO_SUGGEST_OVERRIDE - Success
-- Performing Test HAVE_CXX_WNO_INCONSISTENT_MISSING_OVERRIDE
-- Performing Test HAVE_CXX_WNO_INCONSISTENT_MISSING_OVERRIDE - Failed
-- Performing Test HAVE_CXX_WNO_IMPLICIT_FALLTHROUGH
-- Performing Test HAVE_CXX_WNO_IMPLICIT_FALLTHROUGH - Success
-- Performing Test HAVE_CXX_WNO_TAUTOLOGICAL_COMPARE
-- Performing Test HAVE_CXX_WNO_TAUTOLOGICAL_COMPARE - Success
-- Performing Test HAVE_CXX_WNO_MISSING_PROTOTYPES
-- Performing Test HAVE_CXX_WNO_MISSING_PROTOTYPES - Failed
-- Performing Test HAVE_CXX_WNO_REORDER
-- Performing Test HAVE_CXX_WNO_REORDER - Success
-- Performing Test HAVE_CXX_WNO_UNUSED_RESULT
-- Performing Test HAVE_CXX_WNO_UNUSED_RESULT - Success
-- Performing Test HAVE_CXX_WNO_CLASS_MEMACCESS
-- Performing Test HAVE_CXX_WNO_CLASS_MEMACCESS - Success
CMake Warning (dev) at /usr/share/cmake-3.16/Modules/FindOpenGL.cmake:275 (message):
  Policy CMP0072 is not set: FindOpenGL prefers GLVND by default when
  available.  Run "cmake --help-policy CMP0072" for policy details.  Use the
  cmake_policy command to set the policy and suppress this warning.

  FindOpenGL found both a legacy GL library:

    OPENGL_gl_LIBRARY: /usr/lib/x86_64-linux-gnu/libGL.so

  and GLVND libraries for OpenGL and GLX:

    OPENGL_opengl_LIBRARY: /usr/lib/x86_64-linux-gnu/libOpenGL.so
    OPENGL_glx_LIBRARY: /usr/lib/x86_64-linux-gnu/libGLX.so

  OpenGL_GL_PREFERENCE has not been set to "GLVND" or "LEGACY", so for
  compatibility with CMake 3.10 and below the legacy GL library will be used.
Call Stack (most recent call first):
  cmake/OpenCVFindLibsGUI.cmake:76 (find_package)
  CMakeLists.txt:685 (include)
This warning is for project developers.  Use -Wno-dev to suppress it.

-- Found TBB (cmake): /usr/lib/x86_64-linux-gnu/libtbb.so.2
-- IPPICV: Download: ippicv_2020_lnx_intel64_20191018_general.tgz
-- found Intel IPP (ICV version): 2020.0.0 [2020.0.0 Gold]
-- at: /home/mimi/Pulpit/AAOS_proj3/opencv/build/3rdparty/ippicv/ippicv_lnx/icv
-- found Intel IPP Integration Wrappers sources: 2020.0.0
-- at: /home/mimi/Pulpit/AAOS_proj3/opencv/build/3rdparty/ippicv/ippicv_lnx/iw
-- Could not find OpenBLAS include. Turning OpenBLAS_FOUND off
-- Could not find OpenBLAS lib. Turning OpenBLAS_FOUND off
-- Found Atlas: /usr/include/x86_64-linux-gnu
-- Found Atlas (include: /usr/include/x86_64-linux-gnu, library: /usr/lib/x86_64-linux-gnu/libatlas.so)
-- LAPACK(Atlas): LAPACK_LIBRARIES: /usr/lib/x86_64-linux-gnu/liblapack.so;/usr/lib/x86_64-linux-gnu/libcblas.so;/usr/lib/x86_64-linux-gnu/libatlas.so
-- LAPACK(Atlas): Support is enabled.
-- Performing Test HAVE_CXX_WNO_UNUSED_PARAMETER
-- Performing Test HAVE_CXX_WNO_UNUSED_PARAMETER - Success
-- Performing Test HAVE_CXX_WNO_UNUSED_LOCAL_TYPEDEFS
-- Performing Test HAVE_CXX_WNO_UNUSED_LOCAL_TYPEDEFS - Success
-- Performing Test HAVE_CXX_WNO_SIGN_PROMO
-- Performing Test HAVE_CXX_WNO_SIGN_PROMO - Success
-- Performing Test HAVE_CXX_WNO_TAUTOLOGICAL_UNDEFINED_COMPARE
-- Performing Test HAVE_CXX_WNO_TAUTOLOGICAL_UNDEFINED_COMPARE - Failed
-- Performing Test HAVE_CXX_WNO_IGNORED_QUALIFIERS
-- Performing Test HAVE_CXX_WNO_IGNORED_QUALIFIERS - Success
-- Performing Test HAVE_CXX_WNO_UNUSED_FUNCTION
-- Performing Test HAVE_CXX_WNO_UNUSED_FUNCTION - Success
-- Performing Test HAVE_CXX_WNO_UNUSED_CONST_VARIABLE
-- Performing Test HAVE_CXX_WNO_UNUSED_CONST_VARIABLE - Success
-- Performing Test HAVE_CXX_WNO_SHORTEN_64_TO_32
-- Performing Test HAVE_CXX_WNO_SHORTEN_64_TO_32 - Failed
-- Performing Test HAVE_CXX_WNO_INVALID_OFFSETOF
-- Performing Test HAVE_CXX_WNO_INVALID_OFFSETOF - Success
-- Performing Test HAVE_CXX_WNO_ENUM_COMPARE_SWITCH
-- Performing Test HAVE_CXX_WNO_ENUM_COMPARE_SWITCH - Failed
-- Could NOT find JNI (missing: JAVA_AWT_LIBRARY JAVA_JVM_LIBRARY JAVA_INCLUDE_PATH JAVA_INCLUDE_PATH2 JAVA_AWT_INCLUDE_PATH)
-- VTK is not found. Please set -DVTK_DIR in CMake to VTK build directory, or to VTK install subdirectory with VTKConfig.cmake file
-- Looking for dlerror in dl
-- Looking for dlerror in dl - found
-- Performing Test HAVE_C_WNO_UNDEF
-- Performing Test HAVE_C_WNO_UNDEF - Success
-- ADE: Download: v0.1.1f.zip
-- OpenCV Python: during development append to PYTHONPATH: /home/mimi/Pulpit/AAOS_proj3/opencv/build/python_loader
-- Checking for modules 'libavcodec;libavformat;libavutil;libswscale'
--   No package 'libavcodec' found
--   No package 'libavformat' found
--   No package 'libswscale' found
-- Checking for module 'gstreamer-base-1.0'
--   Found gstreamer-base-1.0, version 1.16.2
-- Checking for module 'gstreamer-app-1.0'
--   Found gstreamer-app-1.0, version 1.16.2
-- Checking for module 'gstreamer-riff-1.0'
--   Found gstreamer-riff-1.0, version 1.16.2
-- Checking for module 'gstreamer-pbutils-1.0'
--   Found gstreamer-pbutils-1.0, version 1.16.2
-- Checking for module 'libdc1394-2'
--   No package 'libdc1394-2' found
-- Caffe:   NO
-- Protobuf:   NO
-- Glog:   NO
-- Checking for module 'freetype2'
--   Found freetype2, version 23.1.17
-- Checking for module 'harfbuzz'
--   Found harfbuzz, version 2.6.4
-- freetype2:   YES (ver 23.1.17)
-- harfbuzz:    YES (ver 2.6.4)
-- Could NOT find HDF5 (missing: HDF5_LIBRARIES HDF5_INCLUDE_DIRS) (found version "")
-- Module opencv_ovis disabled because OGRE3D was not found
-- No preference for use of exported gflags CMake configuration set, and no hints for include/library directories provided. Defaulting to preferring an installed/exported gflags CMake configuration if available.
-- Found installed version of gflags: /usr/lib/x86_64-linux-gnu/cmake/gflags
-- Detected gflags version: 2.2.2
-- Failed to find glog - Could not find glog include directory, set GLOG_INCLUDE_DIR to directory containing glog/logging.h
-- Module opencv_sfm disabled because the following dependencies are not found: Glog/Gflags
-- Checking for module 'tesseract'
--   No package 'tesseract' found
-- Tesseract:   NO
-- Allocator metrics storage type: 'long long'
-- Registering hook 'INIT_MODULE_SOURCES_opencv_dnn': /home/mimi/Pulpit/AAOS_proj3/opencv/modules/dnn/cmake/hooks/INIT_MODULE_SOURCES_opencv_dnn.cmake
-- opencv_dnn: filter out cuda4dnn source code
-- Performing Test HAVE_CXX_WNO_OVERLOADED_VIRTUAL
-- Performing Test HAVE_CXX_WNO_OVERLOADED_VIRTUAL - Success
-- xfeatures2d/boostdesc: Download: boostdesc_bgm.i
-- xfeatures2d/boostdesc: Download: boostdesc_bgm_bi.i
-- xfeatures2d/boostdesc: Download: boostdesc_bgm_hd.i
-- xfeatures2d/boostdesc: Download: boostdesc_binboost_064.i
-- xfeatures2d/boostdesc: Download: boostdesc_binboost_128.i
-- xfeatures2d/boostdesc: Download: boostdesc_binboost_256.i
-- xfeatures2d/boostdesc: Download: boostdesc_lbgm.i
-- xfeatures2d/vgg: Download: vgg_generated_48.i
-- xfeatures2d/vgg: Download: vgg_generated_64.i
-- xfeatures2d/vgg: Download: vgg_generated_80.i
-- xfeatures2d/vgg: Download: vgg_generated_120.i
-- data: Download: face_landmark_model.dat
-- Performing Test HAVE_CXX_WNO_UNUSED_PRIVATE_FIELD
-- Performing Test HAVE_CXX_WNO_UNUSED_PRIVATE_FIELD - Failed
--
-- General configuration for OpenCV 4.3.0-dev =====================================
--   Version control:               4.3.0-344-gc3e8a82c9c
--
--   Extra modules:
--     Location (extra):            /home/mimi/Pulpit/AAOS_proj3/opencv_contrib/modules
--     Version control (extra):     4.3.0-57-g8fc6067b
--
--   Platform:
--     Timestamp:                   2020-05-31T16:49:19Z
--     Host:                        Linux 5.4.0-31-generic x86_64
--     CMake:                       3.16.3
--     CMake generator:             Unix Makefiles
--     CMake build tool:            /usr/bin/make
--     Configuration:               RELEASE
--
--   CPU/HW features:
--     Baseline:                    SSE SSE2 SSE3
--       requested:                 SSE3
--     Dispatched code generation:  SSE4_1 SSE4_2 FP16 AVX AVX2 AVX512_SKX
--       requested:                 SSE4_1 SSE4_2 AVX FP16 AVX2 AVX512_SKX
--       SSE4_1 (17 files):         + SSSE3 SSE4_1
--       SSE4_2 (2 files):          + SSSE3 SSE4_1 POPCNT SSE4_2
--       FP16 (1 files):            + SSSE3 SSE4_1 POPCNT SSE4_2 FP16 AVX
--       AVX (5 files):             + SSSE3 SSE4_1 POPCNT SSE4_2 AVX
--       AVX2 (31 files):           + SSSE3 SSE4_1 POPCNT SSE4_2 FP16 FMA3 AVX AVX2
--       AVX512_SKX (7 files):      + SSSE3 SSE4_1 POPCNT SSE4_2 FP16 FMA3 AVX AVX2 AVX_512F AVX512_COMMON AVX512_SKX
--
--   C/C++:
--     Built as dynamic libs?:      YES
--     C++ standard:                11
--     C++ Compiler:                /usr/bin/c++  (ver 9.3.0)
--       C++ flags (Release):         -fsigned-char -W -Wall -Werror=return-type -Werror=non-virtual-dtor -Werror=address -Werror=sequence-point -Wformat -Werror=format-security -Wmissing-declarations -Wundef -Winit-self -Wpointer-arith -Wshadow -Wsign-promo -Wuninitialized -Winit-self -Wsuggest-override -Wno-delete-non-virtual-dtor -Wno-comment -Wimplicit-fallthrough=3 -Wno-strict-overflow -fdiagnostics-show-option -Wno-long-long -pthread -fomit-frame-pointer -ffunction-sections -fdata-sections  -msse -msse2 -msse3 -fvisibility=hidden -fvisibility-inlines-hidden -O3 -DNDEBUG  -DNDEBUG
--       C++ flags (Debug):           -fsigned-char -W -Wall -Werror=return-type -Werror=non-virtual-dtor -Werror=address -Werror=sequence-point -Wformat -Werror=format-security -Wmissing-declarations -Wundef -Winit-self -Wpointer-arith -Wshadow -Wsign-promo -Wuninitialized -Winit-self -Wsuggest-override -Wno-delete-non-virtual-dtor -Wno-comment -Wimplicit-fallthrough=3 -Wno-strict-overflow -fdiagnostics-show-option -Wno-long-long -pthread -fomit-frame-pointer -ffunction-sections -fdata-sections  -msse -msse2 -msse3 -fvisibility=hidden -fvisibility-inlines-hidden -g -O0 -DDEBUG -D_DEBUG
--     C Compiler:                  /usr/bin/cc
--     C flags (Release):           -fsigned-char -W -Wall -Werror=return-type -Werror=address -Werror=sequence-point -Wformat -Werror=format-security -Wmissing-declarations -Wmissing-prototypes -Wstrict-prototypes -Wundef -Winit-self -Wpointer-arith -Wshadow -Wuninitialized -Winit-self -Wno-comment -Wimplicit-fallthrough=3 -Wno-strict-overflow -fdiagnostics-show-option -Wno-long-long -pthread -fomit-frame-pointer -ffunction-sections -fdata-sections  -msse -msse2 -msse3 -fvisibility=hidden -O3 -DNDEBUG  -DNDEBUG
--     C flags (Debug):             -fsigned-char -W -Wall -Werror=return-type -Werror=address -Werror=sequence-point -Wformat -Werror=format-security -Wmissing-declarations -Wmissing-prototypes -Wstrict-prototypes -Wundef -Winit-self -Wpointer-arith -Wshadow -Wuninitialized -Winit-self -Wno-comment -Wimplicit-fallthrough=3 -Wno-strict-overflow -fdiagnostics-show-option -Wno-long-long -pthread -fomit-frame-pointer -ffunction-sections -fdata-sections  -msse -msse2 -msse3 -fvisibility=hidden -g  -O0 -DDEBUG -D_DEBUG
--     Linker flags (Release):      -Wl,--exclude-libs,libippicv.a -Wl,--exclude-libs,libippiw.a   -Wl,--gc-sections -Wl,--as-needed
--     Linker flags (Debug):        -Wl,--exclude-libs,libippicv.a -Wl,--exclude-libs,libippiw.a   -Wl,--gc-sections -Wl,--as-needed
--     ccache:                      NO
--     Precompiled headers:         NO
--     Extra dependencies:          dl m pthread rt /usr/lib/x86_64-linux-gnu/libGL.so /usr/lib/x86_64-linux-gnu/libGLU.so
--     3rdparty dependencies:
--
--   OpenCV modules:
--     To be built:                 alphamat aruco bgsegm bioinspired calib3d ccalib core cvv datasets dnn dnn_objdetect dnn_superres dpm face features2d flann freetype fuzzy gapi hfs highgui img_hash imgcodecs imgproc intensity_transform line_descriptor ml objdetect optflow phase_unwrapping photo plot python3 quality rapid reg rgbd saliency shape stereo stitching structured_light superres surface_matching text tracking ts video videoio videostab xfeatures2d ximgproc xobjdetect xphoto
--     Disabled:                    world
--     Disabled by dependency:      -
--     Unavailable:                 cnn_3dobj cudaarithm cudabgsegm cudacodec cudafeatures2d cudafilters cudaimgproc cudalegacy cudaobjdetect cudaoptflow cudastereo cudawarping cudev hdf java js matlab ovis python2 sfm viz
--     Applications:                tests perf_tests apps
--     Documentation:               NO
--     Non-free algorithms:         NO
--
--   GUI:
```

```
--     QT:                          YES (ver 5.12.8)
--       QT OpenGL support:         YES (Qt5::OpenGL 5.12.8)
--     GTK+:                        NO
--     OpenGL support:              YES (/usr/lib/x86_64-linux-gnu/libGL.so /usr/lib/x86_64-linux-gnu/libGLU.so)
--     VTK support:                 NO
--
--   Media I/O:
--     ZLib:                        /usr/lib/x86_64-linux-gnu/libz.so (ver 1.2.11)
--     JPEG:                        /usr/lib/x86_64-linux-gnu/libjpeg.so (ver 80)
--     WEBP:                        build (ver encoder: 0x020f)
--     PNG:                         /usr/lib/x86_64-linux-gnu/libpng.so (ver 1.6.37)
--     TIFF:                        /usr/lib/x86_64-linux-gnu/libtiff.so (ver 42 / 4.1.0)
--     JPEG 2000:                   build Jasper (ver 1.900.1)
--     OpenEXR:                     build (ver 2.3.0)
--     HDR:                         YES
--     SUNRASTER:                   YES
--     PXM:                         YES
--     PFM:                         YES
--
--   Video I/O:
--     DC1394:                      NO
--     FFMPEG:                      NO
--       avcodec:                   NO
--       avformat:                  NO
--       avutil:                    NO
--       swscale:                   NO
--       avresample:                NO
--     GStreamer:                   YES (1.16.2)
--     v4l/v4l2:                    YES (linux/videodev2.h)
--
--   Parallel framework:            TBB (ver 2020.1 interface 11101)
--
--   Trace:                         YES (with Intel ITT)
--
--   Other third-party libraries:
--     Intel IPP:                   2020.0.0 Gold [2020.0.0]
--            at:                   /home/mimi/Pulpit/AAOS_proj3/opencv/build/3rdparty/ippicv/ippicv_lnx/icv
--     Intel IPP IW:                sources (2020.0.0)
--               at:                /home/mimi/Pulpit/AAOS_proj3/opencv/build/3rdparty/ippicv/ippicv_lnx/iw
--     Lapack:                      YES (/usr/lib/x86_64-linux-gnu/liblapack.so /usr/lib/x86_64-linux-gnu/libcblas.so /usr/lib/x86_64-linux-gnu/libatlas.so)
--     Eigen:                       YES (ver 3.3.7)
--     Custom HAL:                  NO
--     Protobuf:                    build (3.5.1)
--
--   OpenCL:                        YES (no extra features)
--     Include path:                /home/mimi/Pulpit/AAOS_proj3/opencv/3rdparty/include/opencl/1.2
--     Link libraries:              Dynamic load
--
--   Python 3:
--     Interpreter:                 /usr/bin/python3 (ver 3.8.2)
--     Libraries:                   /usr/lib/x86_64-linux-gnu/libpython3.8.so (ver 3.8.2)
--     numpy:                       /home/mimi/.local/lib/python3.8/site-packages/numpy/core/include (ver 1.18.3)
--     install path:                lib/python3.8/dist-packages/cv2/python-3.8
--
--   Python (for build):            /usr/bin/python2.7
--
--   Java:
--     ant:                         NO
--     JNI:                         NO
--     Java wrappers:               NO
--     Java tests:                  NO
--
--   Install to:                    /usr/local
-- -----------------------------------------------------------------
--
-- Configuring done
-- Generating done
-- Build files have been written to: /home/mimi/Pulpit/AAOS_proj3/opencv/build
```