Hyperyon Dual Stream

Linux Installation Manual

e-con Systems

Your Product Development Partner

Version 1.0

e-con Systems

7/3/2020

Disclaimer

e-con Systems reserves the right to edit/modify this document without any prior intimation of whatsoever.

Contents

Introduction to Hyperyon Dual Stream

Hyperyon Dual Stream is an application which explores the feature of dual streaming of eCAM22_USB. It uses OpenCV libraries to communicate with camera. APIs introduced in the OpenCV can be supported with all e-con Systems cameras.

This document helps to install the OpenCV (with e-con Systems patch) in Linux and build a sample code to access the Hyperyon dual stream camera with OpenCV.

Prerequisites

The prerequisites are as follows:

- Download OpenCV from this https://github.com/opencv/opencv link or run the following command to download the official OpenCV release using wget.

```
$ cd ~

$ wget -O opencv.zip \
https://github.com/opencv/opencv/archive/<3.3.1 or
3.4.1 >.zip
```

- GCC 4.4
- CMake 2.8.7 or higher
- GTK+2.x or higher, including headers (libgtk2.0-dev)
- Library packages: libjpeg-dev, libpng-dev, libtiff-dev, libjasper-dev, libavcodecdev, libavformat-dev, libswscale-dev, libv4l-dev, libxvidcore-dev, libx264-dev, libgtk-3-dev, libatlas-base-dev, gfortran

Description

The steps described in this document are tested on Ubuntu 18.04 (Bionic Beaver). OpenCV must work on any other relatively modern version of Linux OS.

Building OpenCV

The Hyperyon Dual Stream is a sample command line application used to demonstrate the feature dual streaming of e-CAM22_USB with OpenCV APIs. The steps to build OpenCV are as follows:

Step 1. Installing Dependencies
Step 2. Configuring OpenCV

Step 1 - Installing Dependencies

The below table lists the dependencies to be installed for using OpenCV.

Table 1: Installing Dependencies

Dependencies	Commands
Some general development libraries	`$ sudo apt-get install build-essential make cmake cmake-qt-gui g++ unzip pkg-config`
Video4Linux Camera development libraries	`$ sudo apt-get install libv4l-dev`
OpenGL development libraries for creating graphical windows	`$ sudo apt-get install libglew-dev`
GTK development libraries for creating graphical windows	`$ sudo apt-get install libgtk-3-dev`
Udev development libraries for accessing device information	`$ sudo apt-get install libudev-dev`
Libav video input or output development libraries	`$ sudo apt-get install libavformat-dev libavutil-dev libswscale-dev libavcodec-dev libavcodec-ffmpeg-extra56 libavformat-ffmpeg56 libavutil-ffmpeg54 libswscale-ffmpeg3 libdc1394-* libjpeg-dev libpng-dev libtiff-dev libjasper-dev libxvidcore-dev libx264-dev`
Eigen3 math development libraries	`$ sudo apt-get install libeigen3-dev`

Step 2 - Configuring OpenCV

The steps to configure OpenCV are as follows:

- Replace the **videoio** folder inside the downloaded /opencv-<3.3.1 or 3.4.1> /modules directory with the folder provided by e-con Systems.
- Run the following commands to navigate to the downloaded opencv-<3.3.1 or 3.4.1> directory, to build and install OpenCV.

```
$ mkdir release
$ cd release
$ cmake -D CMAKE_BUILD_TYPE=RELEASE \
-D CMAKE_INSTALL_PREFIX=/usr/local \
-D WITH_CUDA=OFF \
-D INSTALL_PYTHON_EXAMPLES=OFF \
-D OPENCV_ENABLE_NONFREE=ON \
-D BUILD_EXAMPLES=ON \
-D WITH_GPHOTO2=OFF \
-D BUILD_TESTS=OFF \
-D WITH_VTK=OFF \
-D BUILD_PERF_TESTS=OFF ..
```

- Run the following command to build and install the OpenCV libraries in the **/usr/local/lib/** location.

```
$ sudo make -j4 && sudo make install
```

Building Sample Code

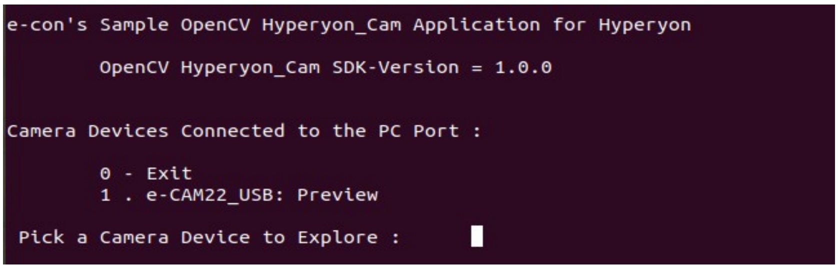
This section describes about how to build the sample code.

The steps to run sample application are as follows:

1. Open Hyperyon Dual Stream sample command line application from source folder.
2. Run the following commands to build the sample code.

```
$ sudo make
$ sudo ./HyperyonCam
```

The devices connected to the PC will be displayed as below shown.



```
e-con's Sample OpenCV Hyperyon_Cam Application for Hyperyon

        OpenCV Hyperyon_Cam SDK-Version = 1.0.0

Camera Devices Connected to the PC Port :

        0 - Exit
        1 . e-CAM22_USB: Preview

 Pick a Camera Device to Explore :
```

Using Prebuilt Binary

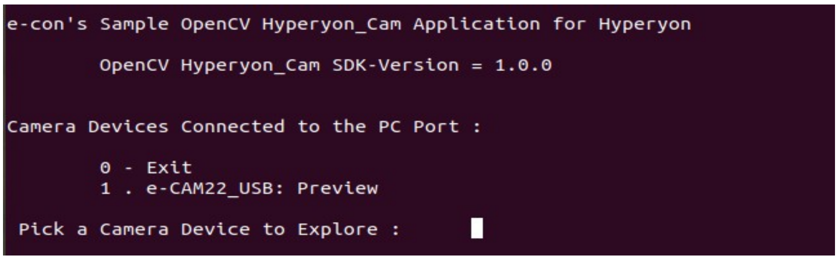
This section describes about how to build the sample code.

The steps to run sample application are as follows:

1. Download the Hyperyon Dual Stream application package.
2. Navigate to the source folder in the corresponding prebuilt library directory.
3. Run the following command to run the sample code.

```
$ sudo LD_LIBRARY_PATH=”./” ./HyperyonCam
```

The devices connected to the PC will be displayed as below shown.



```
e-con's Sample OpenCV Hyperyon_Cam Application for Hyperyon

        OpenCV Hyperyon_Cam SDK-Version = 1.0.0


Camera Devices Connected to the PC Port :

        0 - Exit
        1 . e-CAM22_USB: Preview

 Pick a Camera Device to Explore :
```

Troubleshooting

In this section, you can view the list of commonly occurring issues and their troubleshooting steps.

CMake Error: The source directory does not appear to contain CMakeLists.txt

1. Run cmake-gui from the terminal
   - Provide the source and release folder path of the OpenCV project
   - Configure and generate the solution

Compiling command line application, libudev.so error adding symbols, DSO missing from command line

Include the ludev dependency in the compilation command.

For example, CMAKE_CXX_FLAGS=”-ludev”

Error loading libopencv_world.so

Run the following command (in order to take prebuilt lib path should be specified).

```
sudo LD_LIBRARY_PATH=<path_to_the_libopencv_world.so>
./HyperyonCam
```

Instead, if you wish to add the path to libopencv_world.so, permanently to the library path. Go to **/etc/ld.so.conf.d/**. Create a config file named opencv.conf, mention the path inside the config file and run the following command.

```
$ sudo ldconfig
```

Undefined reference to main:

1. Replace **videoio** module freshly.
2. Delete all opencv folders present in **/usr/local/include/** location.
3. Rebuild opencv again.

   If problem still exist, open **cmake gui** by running the following command.

   ```
   $ cmake-gui
   ```

   - Uncheck OpenCL, Gstreamer and opencv test flags.

Support

Contact Us

If you need any support on Hyperyon Dual Stream application, please contact us using the Live Chat option available on our website - https://www.e-consystems.com/

Creating a Ticket

If you need to create a ticket for any type of issue, please visit the ticketing page on our website - https://www.e-consystems.com/create-ticket.asp

RMA

To know about our Return Material Authorization (RMA) policy, please visit the RMA Policy page on our website - https://www.e-consystems.com/RMA-Policy.asp

General Product Warranty Terms

To know about our General Product Warranty Terms, please visit the General Warranty Terms page on our website - https://www.econsystems.com/warranty.asp

Revision History

Rev	Date	Description	Author
1.0	03-July-2020	Initial Draft	M Vishnu Murali