

Cartoon Filter that demonstrates OpenCV-CL Inter-operations

1 Overview

1.1 Location `$<APPSDKSamplesInstallPath>\samples\opencv\`

1.2 How to Run See the *Getting Started* guide for how to build samples. You first must compile the sample.

Use the command line to change to the directory where the executable is located. The pre-compiled sample executable is at `$<APPSDKSamplesInstallPath>\samples\opencv\bin\x86\` for 32-bit builds, and `$<APPSDKSamplesInstallPath>\samples\opencv\bin\x86_64\` for 64-bit builds.

Type the following command(s).

1. `CartoonFilter`
This command runs the program with the default options.
2. `CartoonFilter -h`
This command prints the help file.

1.3 Command Line Options Table 1 lists, and briefly describes, the command line options.

Table 1 Command Line Options

Short Form	Long Form	Description
-h	--help	Shows all command options and their respective meaning.
-q	--quiet	Quiet mode. Suppresses most text output.
-e	--verify	Verify results against CPU reference implementation.
-t	--timing	Print timing.
-v	--version	Print AMD APP SDK version.
-f	--imageFile	Option to provide the input image file. The default image file is <code>lena.jpg</code> .
-i	--iterations	Number of iterations for kernel execution.

2 Introduction

This sample uses the cartoon filter as an example to show the different forms of inter-operations between OpenCV, OpenCV-CL, and raw OpenCL kernels. An input image is converted to a cartoon-like image.

3 Implementation Details

The following pipeline shows various data transfer inter-operations (interop) as well as kernel interops. Figure 1 illustrates the direction of the data flow.

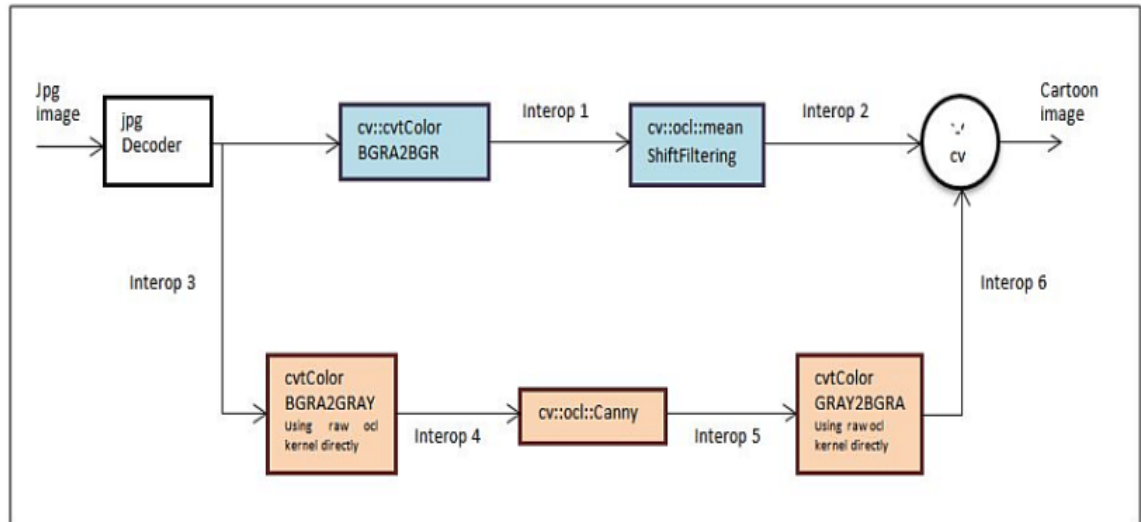


Figure 1 Interop data flow

The flow in Figure 1 uses the following interop legend:

1. Interop 1: OpenCV to OpenCV-CL
2. Interop 2: OpenCV-CL to OpenCV
3. Interop 3: OpenCV to Raw OpenCL kernel
4. Interop 4: Raw OpenCL kernel to OpenCV-CL
5. Interop 5: OpenCV-CL to Raw OpenCL kernel
6. Interop 6: Raw OpenCL kernel to OpenCV

The pipeline consists of two stages. In Stage 1, the input image is passed to OpenCV's convert color routine to convert from a BGRA image to a BGR image. The BGR image in OpenCV format is converted to the OpenCV-CL format and passed to `cv::ocl::meanShiftFiltering`. The output is a mean shift filtered by a spatial window radius of 10 and a color window radius of 30.

In Stage 2, the input image is color converted again from BGRA to a gray image, but this time using the pure OpenCL kernel found in the `kernel.cl` file. The `RGB2Gray` caller initializes channels, global threads, local threads, build options, and arguments to the `RGB2Gray` OpenCL kernel. The kernel is provided as a string to the `extern` variable declared. The initialized values, arguments, and the kernel are then passed to the `openCLExecuteKernelInterop` function, where the OpenCV-OpenCL interop happens. The color-converted gray image is then passed on to `cv::ocl::Canny`, which does not require any conversions across OpenCL and OpenCV data structures. The Canny output is again color-converted using the pure OpenCL kernel, `Gray2RGB`. Finally, the output is subtracted from the `meanShiftFiltered` image to obtain the cartooned image.

Contact

Advanced Micro Devices, Inc.
One AMD Place
P.O. Box 3453
Sunnyvale, CA, 94088-3453
Phone: +1.408.749.4000

For AMD Accelerated Parallel Processing:
URL: developer.amd.com/appsdk
Developing: developer.amd.com/
Support: developer.amd.com/appsdksupport
Forum: developer.amd.com/openclforum



The contents of this document are provided in connection with Advanced Micro Devices, Inc. ("AMD") products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. The information contained herein may be of a preliminary or advance nature and is subject to change without notice. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property rights is granted by this publication. Except as set forth in AMD's Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD's products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD's product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

Copyright and Trademarks

© 2013 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, ATI, the ATI logo, Radeon, FireStream, and combinations thereof are trademarks of Advanced Micro Devices, Inc. OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos. Other names are for informational purposes only and may be trademarks of their respective owners.