



OpenCV-OpenCL

开发教程

V1.0

修定历史记录:

日期	版本	说明	作者
2018.10	Version<1.0>	文件创建	Wang

Note: 任何修改操作请在上述文档中备注说明。

目录

一 OpenCV-OpenCL 简介.....	4
二 OpenCL 加载.....	4
三 OpenCV-OpenCL 实例.....	5

一 OpenCV-OpenCL 简介

OpenCV 是一个跨平台的计算机视觉处理库，全称 Open Source Computer Vision。由英特尔公司发起并参与开发，以 BSD 许可证授权发行，可以在商业和研究领域中免费使用。OpenCV 可用于开发实时的图像处理、计算机视觉以及模式识别程序。

本文主要讲解版本是 OpenCV3.1.具体函数库参考连接：

<https://docs.opencv.org/3.1.0/#gsc.tab=0>

OpenCV3.1 架构添加新的概念 transparent API (T-API)，允许在支持的硬件加速器可用时无缝加载 OpenCL。处理器 SDK 提供的 OpenCV3.1 将一些算法通过 OpenCL 加载到 C66x DSP 上。

二 OpenCL 加载

OpenCV3.1 支持大约 200 多个 OpenCL kernels，可以优化不同模块中的关键功能。通过 T-API 加载的 OpenCL kernel 由 UMat 数据结构启用，取代传统的 Mat 数据结构。UMat 尽可能使用 OpenCL 去分配内存，保持与 Mat 数据结构的向后兼容。

Code sample

Regular CPU code

```
// initialization
VideoCapture vcap(...);
CascadeClassifier fd("haar_ff.xml");
Mat frame, frameGray;
vector faces;
for(;;){
    // processing loop
    vcap >> frame;
    cvtColor(frame, frameGray, BGR2GRAY);
    equalizeHist(frameGray, frameGray);
    fd.detectMultiScale(frameGray, faces, ...);
    // draw rectangles ...
    // show image ...
}
```

OpenCL-aware code OpenCV-2.x

```
// initialization
VideoCapture vcap(...);
ocl::OclCascadeClassifier fd("haar_ff.xml");
ocl::oclMat frame, frameGray;
Mat frameCpu;
vector faces;
for(;;){
    // processing loop
    vcap >> frameCpu;
    frame = frameCpu;
    ocl::cvtColor(frame, frameGray, BGR2GRAY);
    ocl::equalizeHist(frameGray, frameGray);
    fd.detectMultiScale(frameGray, faces, ...);
    // draw rectangles ...
    // show image ...
}
```

OpenCL-aware code OpenCV-3.x

```
// initialization
VideoCapture vcap(...);
CascadeClassifier fd("haar_ff.xml");
UMat frame, frameGray;
vector faces;
for(;;){
    // processing loop
    vcap >> frame;
    cvtColor(frame, frameGray, BGR2GRAY);
    equalizeHist(frameGray, frameGray);
    fd.detectMultiScale(frameGray, faces, ...);
    // draw rectangles ...
    // show image ...
}
```

OpenCV3.1 使能加载 OpenCL， 需要设置环境变量 OPENCV_OPENCL_DEVICE

执行如下命令设置环境变量：

```
OPENCV_OPENCL_DEVICE=' TI AM57:ACCELERATOR:TI Multicore C66 DSP'
```

如果上述环境变量未定义，OpenCV 将不会初始化 OpenCL。此外，用户也可以使用库函数 `ocl::setUseOpenCL(true)` 或 `ocl::setUseOpenCL(false)` 去使能/禁用 OpenCL

三 OpenCV-OpenCL 实例

实例源码位于 `JiangNiu-demo/OpenCV/SobelOpencl.cpp`

3.1 实例执行步骤

1 参考文档《IndustriPi 用户指南》，启动板卡，然后将 USB 摄像头连接到板卡 USB 接口

2 开发板运行如下命令，启动 SobelOpencl 实例

```
root@am57xx-evm:~# export OPENCV_OPENCL_DEVICE='TI AM57:ACCELERATOR:TI Multicore C66 DSP'
root@am57xx-evm:~# /opt/JiangNiu-demo/SobelOpencl
```

实例结果：



SobelOpencl 实例由于使用 OpenCL 进行加速，CPU 使用率只有 26%，明显低于无 OpenCL 加速的程序，详细对比如下图所示：

1253	root	20	0	494524	55260	41456	S	26.0	3.0	1:19.48	SobelOpencl
1115	root	20	0	116768	67168	65676	S	10.9	3.6	0:32.90	weston
976	root	-51	0	0	0	0	S	8.2	0.0	0:29.15	irq/432-xh+
1336	root	20	0	3016	1704	1328	R	1.0	0.1	0:01.00	top
80	root	20	0	0	0	0	S	0.7	0.0	0:02.48	kworker/1:1
3	root	20	0	0	0	0	S	0.3	0.0	0:00.48	ksoftirqd/0
4	root	-2	0	0	0	0	S	0.3	0.0	0:00.53	ktimersoft+
1275	root	20	0	481808	57192	40728	R	87.2	3.1	2:52.32	Sobel
1112	root	20	0	120312	69740	65700	R	30.6	3.7	1:06.24	weston
843	root	-51	0	0	0	0	S	14.8	0.0	0:31.40	irq/432-xhci-hc
1086	root	-51	0	0	0	0	S	1.6	0.0	0:03.83	irq/333-SGX ISR
245	root	20	0	0	0	0	S	1.0	0.0	0:01.12	kworker/u4:2
1289	root	20	0	3016	1704	1320	R	1.0	0.1	0:02.29	top
1245	root	20	0	0	0	0	S	0.7	0.0	0:00.53	kworker/1:4

注意：

OpenCV 实例编译指令如下所示：

```
cd ~/ti-processor-sdk-linux-rt-am57xx-evm-03.03.00.04/JiangNiu-demo/OpenCV
```

```
make
```