

6、TensorRT 板载摄像头实时图像识别教程

准备工作：

开始之前，确定已经完成教程 4、 5 的所有步骤，能测试简单的例子。

1、进入执行的目录

假如你在 jetson-inference 目录下请执行以下目录。

```
cd build/aarch64/bin/
```

```
yahboom@yahboom-desktop:~/yahboom/jetson-inference/build/aarch64/bin$ ls
camera-viewer      detectnet-console.py  imagenet-console     segnet-batch.py      trt-bench
camera-viewer.py   gl-display-test       imagenet-console.py  segnet-batch.sh      trt-console
cuda-from-numpy.py gl-display-test.py    images               segnet-camera        v4l2-console
cuda-to-numpy.py   homography-camera     my-detection.py      segnet-camera.py     v4l2-display
detectnet-camera   homography-console    my-recognition.py    segnet-console
detectnet-camera.py imagenet-camera        networks             segnet-console.py
detectnet-console  imagenet-camera.py    output.jpg           superres-console
```

2、执行图像识别命令

这个时候最好能通过桌面执行，否则可能看不到摄像头的界面，或者通过 VNC 远程桌面连接。

进入到 jetson-inference/ aarch64 / bin 目录下：

实时图像识别演示位于 jetson-inference/ aarch64 / bin 中并被调用 `imagenet-camera`。它在实时摄像机流上运行，并根据用户参数，使用 TensorRT 加载 googlenet 或 alexnet。

```
$ ./imagenet-camera googlenet          #使用 googlenet 运行
```

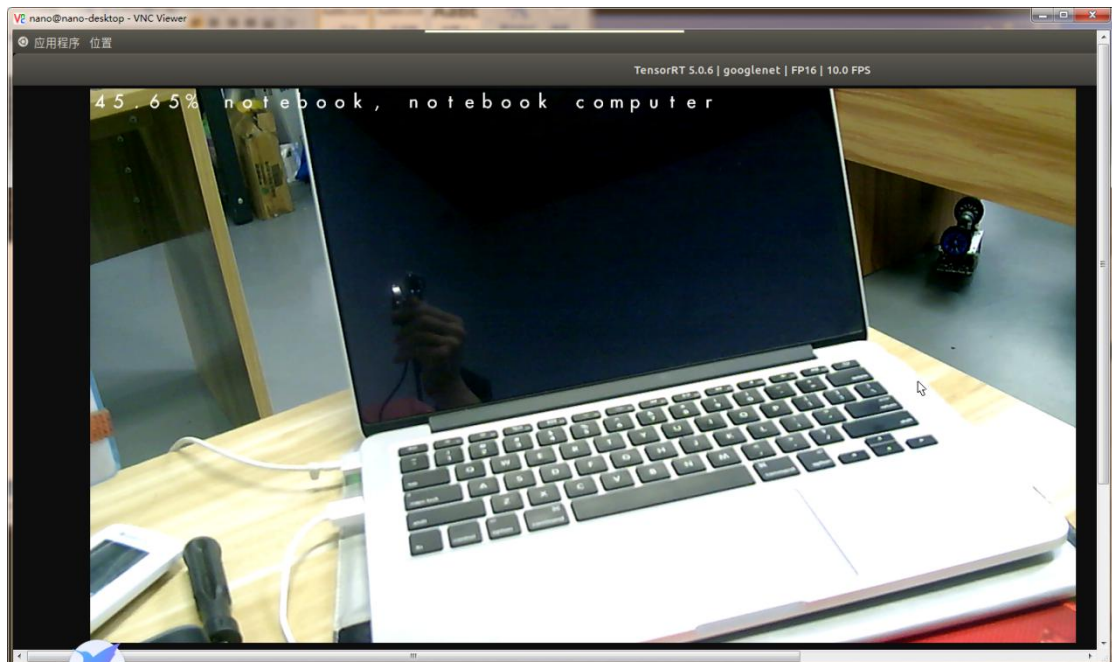
```
$ ./imagenet-camera alexnet            #使用 alexnet 运行
```

每秒帧数（FPS），来自视频的分类对象名称和分类对象的置信度被打印到 OpenGL 窗口标题栏。默认情况下，应用程序可以识别多达 1000 种不同类型的对象，因为 Googlenet 和 Alexnet 是在包含 1000 类对象的 ILSVRC12 ImageNet 数据库上进行培训的。1000 种类型对象的名称映射，可以在 repo 下找到 `data/networks/ilsrvrc12_synset_words.txt`

```

nano@nano-desktop: ~/jetson-inference/build/aarch64/bin
File Edit View Search Terminal Help
class 0894 - 0.011559 (wardrobe, closet, press)
class 0905 - 0.148682 (window shade)
imagenet-camera: 17.79785% class #519 (crate)
class 0421 - 0.029892 (bannister, banister, balustrade, balusters, handrail)
class 0453 - 0.022903 (bookcase)
class 0478 - 0.015747 (carton)
class 0498 - 0.052887 (cinema, movie theater, movie theatre, movie house, picture palace)
class 0519 - 0.102295 (crate)
class 0526 - 0.026581 (desk)
class 0527 - 0.013321 (desktop computer)
class 0553 - 0.043823 (file, file cabinet, filing cabinet)
class 0598 - 0.030716 (home theater, home theatre)
class 0664 - 0.016571 (monitor)
class 0713 - 0.014450 (photocopier)
class 0743 - 0.021194 (prison, prison house)
class 0745 - 0.036469 (projector)
class 0789 - 0.010567 (shoji)
class 0799 - 0.016373 (sliding door)
class 0818 - 0.133911 (spotlight, spot)
class 0894 - 0.010864 (wardrobe, closet, press)
class 0905 - 0.132935 (window shade)
imagenet-camera: 13.39111% class #818 (spotlight, spot)

```



这样识别到物体就会在上面显示物体英文名称，百分数就是匹配百分比。

注意： 安装新摄像头外壳会出现视频画面颠倒，可用以下方法设置修改：

```

$ cd ~/jetson-inference/utils/camera/
$ gedit gstCamera.cpp
...

```

- Change the folloing code:

```
```bash
#if NV_TENSORRT_MAJOR > 1 && NV_TENSORRT_MAJOR < 5 // if
JetPack 3.1-3.3 (different flip-method)
const int flipMethod = 0; // Xavier (w/TRT5) camera is mounted inverted
#else
const int flipMethod = 2;
#endif
```

改成以下:

```
#if NV_TENSORRT_MAJOR > 1 && NV_TENSORRT_MAJOR < 5 // if
JetPack 3.1-3.3 (different flip-method)
const int flipMethod = 0; // Xavier (w/TRT5) camera is mounted inverted
#else
const int flipMethod = 0; //2 变成 0
#endif
```
```

Build the Code:

```
```bash
$ cd ~/jetson-inference/build/
$ make
$ sudo make install
```