docker中运行目标检测

jetson-inference docker环境介绍

- 缺少的模型文件，可以 download-models.sh 中找到下载。

- 下载 SSD-Mobilenet-v2.tar.gz
- 放到
- 宿主机：~/jetson-inference/data/networks
- tar xzvf SSD-Mobilenet-v2.tar.gz

目标检测模型

- 进入docker容器
- docker run -it --rm --runtime nvidia -v ~/jetson-inference/data:/jetson-inference/data 7a7d343029a2 /bin/bash

- docker容器内
- cd /jetson-inference/build/aarch64/bin

- 目标识别
- ./detectnet --network=ssd-mobilenet-v2 images/peds_0.jpg images/test/output.jpg

目标检测结果

训练自己的目标检测模型

- 1. 拍摄自己的对象，标注
- 2.训练
- 3.推理

docker中如何查看宿主机摄像头

- CSI 摄像头：使用 “-v /tmp/argus_socket:/tmp/argus_socket --device /dev/video0” 方式进行映射

- USB 摄像头：使用 “-v /dev/video1:/dev/video1 –device /dev/video1” 方式进行映射

实例:

- csi摄像头
- docker run -it --rm --runtime nvidia -v ~/jetson-inference/data/:/jetson-inference/data/ -v /tmp/argus_socket:/tmp/argus_socket --device /dev/video0 7a7d343029a2 /bin/bash

csi docker中测试摄像头代码

- import cv2
- pipeLine = "nvarguscamerasrc sensor-id=0 !video/x-raw(memory:NVMM), width=(int)1920, height=(int)1080, framerate=(fraction)30/1 ! nvvidconv flip-method=0 ! video/x-raw, width=(int)960, height=(int)540, format=(string)BGRx ! videoconvert ! video/x-raw, format=(string)BGR ! appsink"
- video_capture = cv2.VideoCapture(pipeLine, cv2.CAP_GSTREAMER)
- ret_val, frame = video_capture.read()
- cv2.imwrite("csiCameraDocker.jpg", frame)
- video_capture.release()

usb摄像头映射到docker中

- docker run -it --rm --runtime nvidia -v ~/jetson-inference/data/:/jetson-inference/data/ -v /dev/video1:/dev/video1 --device /dev/video1 7a7d343029a2 /bin/bash

usb docker中测试摄像头代码

- import cv2
- video_capture = cv2.VideoCapture(1)
- ret_val, frame = video_capture.read()
- cv2.imwrite("usbCameraDocker.jpg", frame)
- video_capture.release()

如何在docker 运行带图形化界面的程序?

- --network host -e DISPLAY=$DISPLAY -v /tmp/.X11-unix/:/tmp/.X11-unix -v /etc/enctune.conf:/etc/enctune.conf

- 实例:
- 先运行
- xhost +
- 在运行
- docker run -it --rm --runtime nvidia -v ~/jetson-inference/data/:/jetson-inference/data/ -v /dev/video1:/dev/video1 --device /dev/video1 --network host -e DISPLAY=$DISPLAY -v /tmp/.X11-unix/:/tmp/.X11-unix -v /etc/enctune.conf:/etc/enctune.conf 7a7d343029a2 /bin/bash