

Jetson nano从烧录系统到DeepStream+TensorRT+yolov5检测CSI摄像头视频

学习 2021-06-24 22:56 3074阅读 · 48喜欢 · 29评论



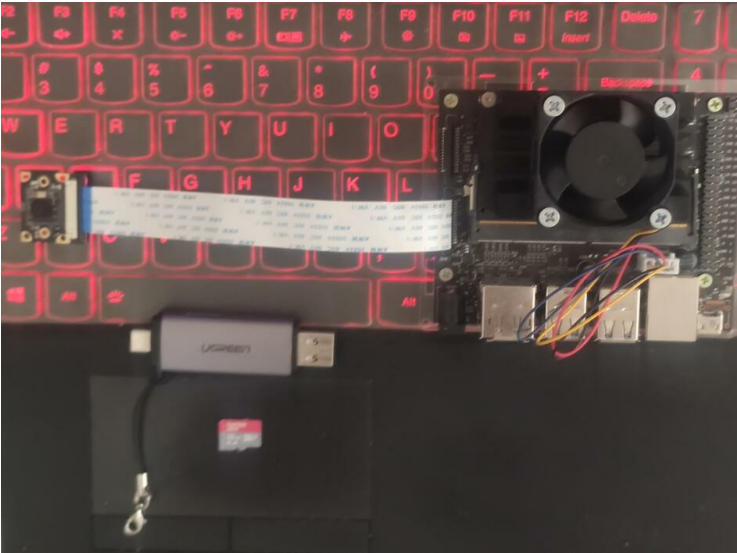
才疏学浅的小佬鼠
 粉丝：746 文章：4

+ 关注

目录

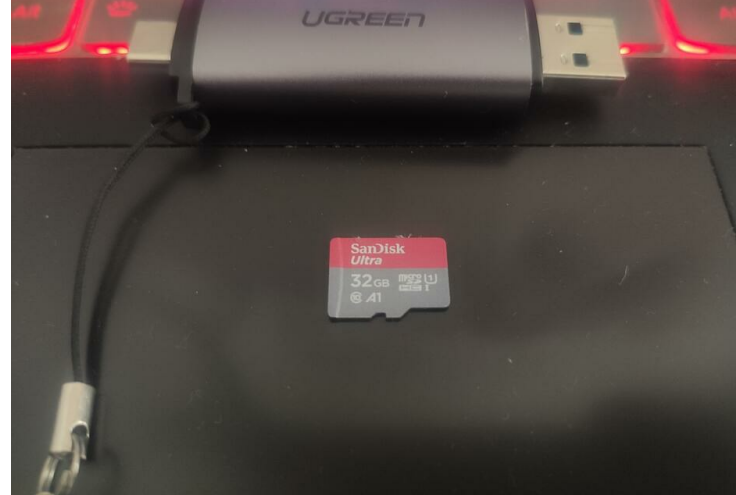
- 一、Jetson nano 镜像下载 & 系统烧录
- 二、Add swap RAM & Auto Fan
- 三、cuda & darknet
- 四、git clone & make darknet 源码 推理 yolov4-tiny
- 五、torch & torchvision 安装 推理yolov5s
- 六、TensorRT make & 推理测试
- 七、DeepStream安装 & yolov5检测CSI摄像头视频

准备



1. Jetson nano硬件【开发套件+CSI摄像头+WiFi网卡（本人用的是USB无线网卡）】

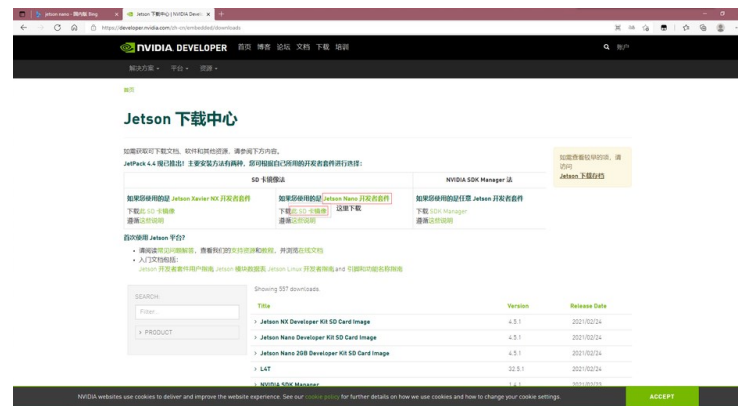




2. USB3.0读卡器和高速SD卡

一、Jetson nano 镜像下载 & 系统烧录

1. Jetson nano官网镜像下载地址: <https://developer.nvidia.com/zh-cn/embedded/downloads>



2. 将下载好的镜像解压烧录

这里可移步官方中文教程: <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#write>





SD Memory Card Formatter下载地址:

https://www.sdcard.org/downloads/formatter_4/eula_windows/

百度网盘下载链接:

链接:<https://pan.baidu.com/s/1B3heluW2ttwm1uMF-vr20Q>

提取码:10b2



Etcher下载地址: <https://www.balena.io/etcher>

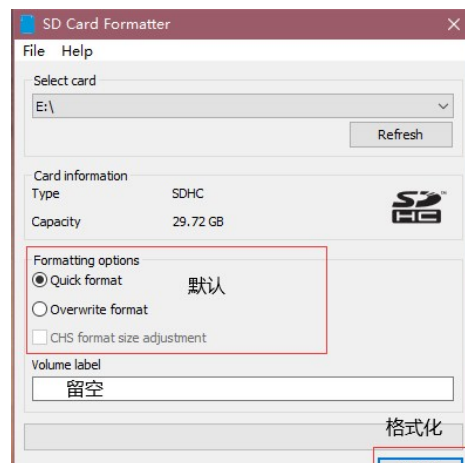
百度网盘下载链接:

链接:https://pan.baidu.com/s/1C4P0mpWih0UDA6h_rmXkzw

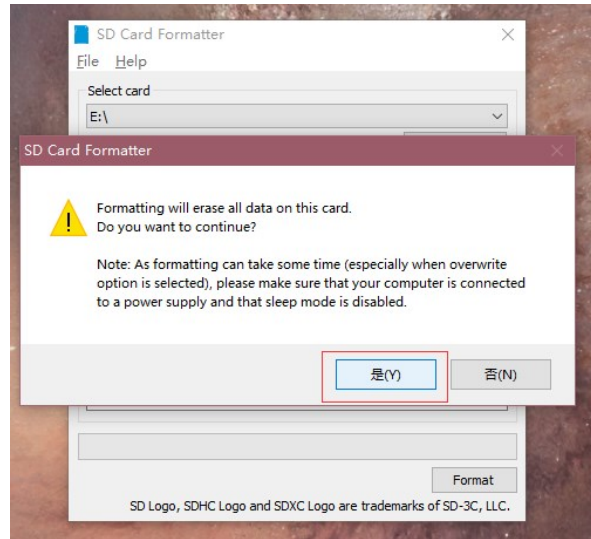
提取码:cokg

3. 格式化SD卡:

1. 下载、安装并启动 Windows 版 SD Memory Card Formatter。



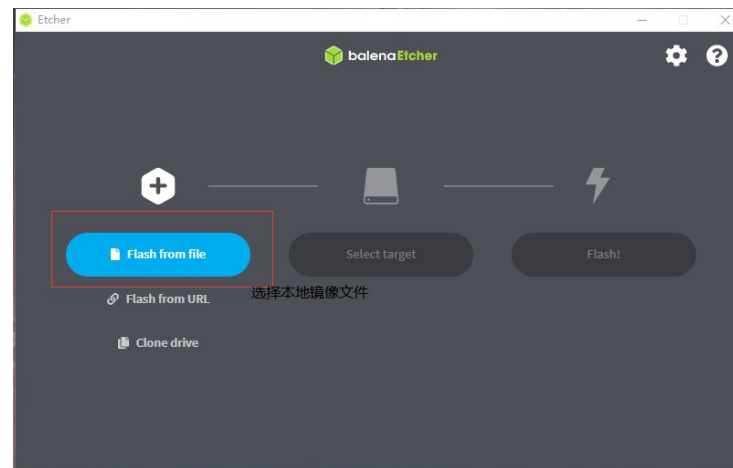
2. 选择卡驱动器、选择“Quick format”（快速格式化），将“Volume label”（卷标）留空。



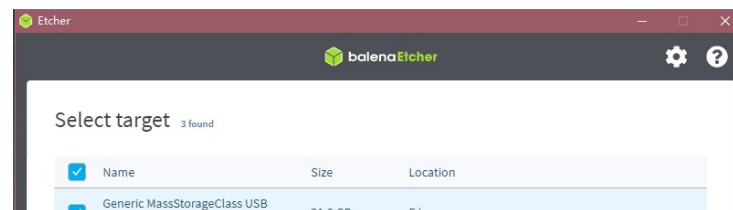
3. 单击“Format”（格式化）开始格式化，然后在警告对话框中单击“Yes”（是）

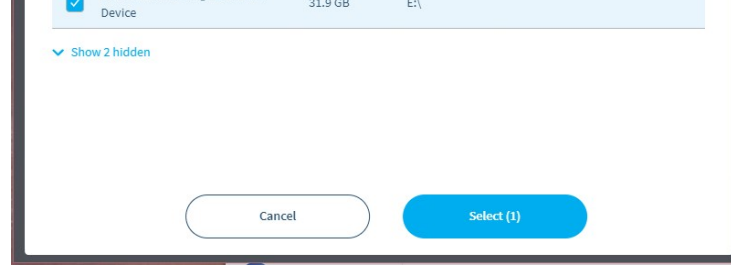
4. 烧录镜像:

1. 下载、安装并启动 Etcher。

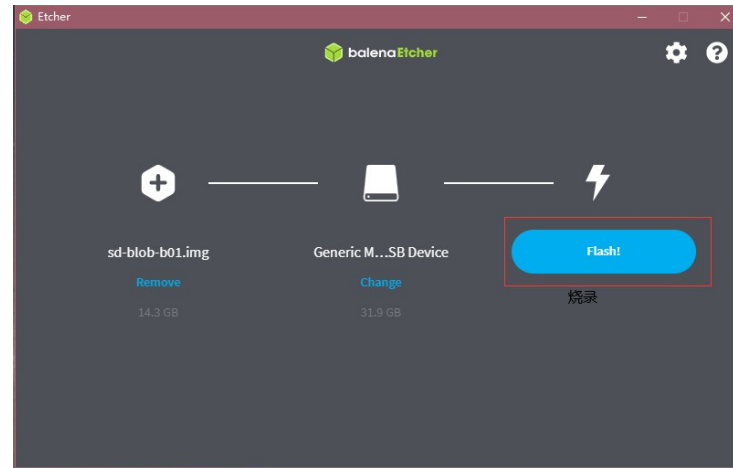


2. 单击“Select image”（选择镜像），然后选择先前下载的压缩镜像文件。

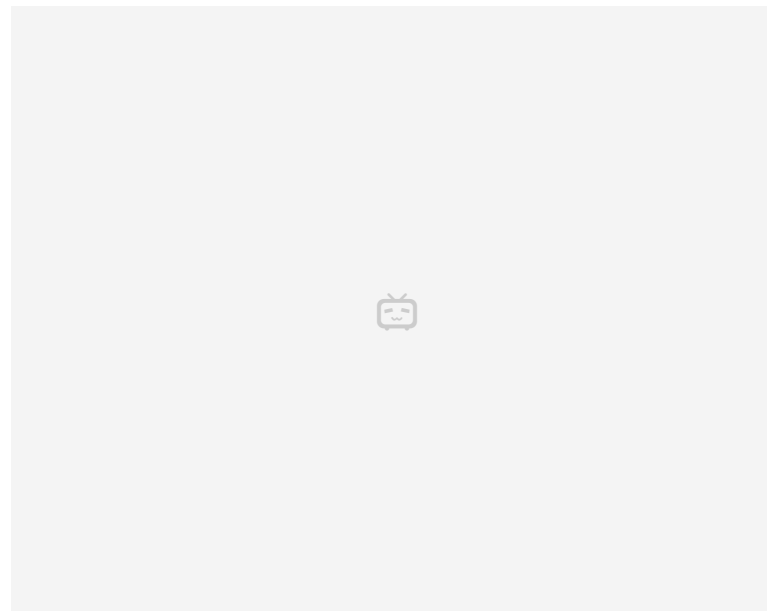




3. 单击“Select drive”（选择驱动器），并选择正确设备。



4. 单击“Flash!”（闪存！）。如果 microSD 卡通过 USB3 连接，Etcher 写入和验证图像约需要 10 分钟。



5. Etcher 操作完成后，Windows 可能会提示您它不知如何读取 SD 卡。此时只需单

击“Cancel”（取消），然后弹出 microSD 卡即可。

6. 安装和首次启动配置

这里可移步官方中文教程: <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#setup>





2. 语言建议选成English（绝对路径含中文都是坑）



3. 键盘布局这里选成Chinese



4.我使用的是无线usb网卡连接wifi，没有网卡用网线或者不联网



5.时区这里选择上海（点一下中国的版图）



6.填写自己的用户名和密码



7.这个先默认即可（稍后会讲）



8.我这里用的是DC5V4A的电源，选MAXN



8.等待设置完成，重启进入桌面



9.安装系统已经完成!



10.进入桌面

二、Add swap RAM & Auto Fan

因为Jetson nano的内存只有4GB，且与显存共用，后期编译torchvision，生成TensorRT推理引擎文件不够用，这里通过增加swap内存的方法来解决这个问题。用到的是<https://blog.csdn.net/watershade2010/article/details/109630135>这位博主的方法。



1. `sudo vim /etc/systemd/nvzramconfig.sh`

按"`i`"进入输入模式，如图所示，修改`mem=$((("${totalmem}" / 2 / "${NRDEVICES}") * 1024))`为`mem=$((("${totalmem}" * 2 / "${NRDEVICES}") * 1024))`

按"Esc"输入`:.wq`保存退出，重启板子



修改前为1.9GiB



修改后为7.7GiB

2. 自动风扇控制，这里使用<https://github.com/Pyrestone/jetson-fan-ctl.git>这个脚本，



1. 打开终端 `git clone https://github.com/Pyrestone/jetson-fan-ctl.git` # 克隆到本地
2. `cd jetson-fan-ctl.git`
3. `sudo ./install.sh` # 自动安装
4. `sudo service automagic-fan restart` # 设置开机自启

三、cuda & darknet



1. `vim .bashrc`
2. 在末尾加上

```
export PATH=/usr/local/cuda-10.2/bin${PATH:+:${PATH}}
export LD_LIBRARY_PATH=/usr/local/cuda-10.2/lib64\
${LD_LIBRARY_PATH:+:${LD_LIBRARY_PATH}}
```
3. 按"**Esc**"输入"**:wq**"保存退出



4. 输入`nvcc -V` 即可查看cuda版本为10.2.89

四、git clone & make darknet 源码 推理 yolov4-tiny

1. 下载yolov4源码: `git clone https://github.com/AlexeyAB/darknet.git`



2. 按照下图修改[Makefile](#)文件

GPU=1 CUDNN=1 OPENCV=1 LIBSO=1



3. 按"**Esc**"输入`:.wq`保存退出，输入**make -j**编译



4. 编译结束，输入`./darknet`有如图所示的输出，编译成功



5. 输

入 `wget https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v4_pre/yolov4-tiny.weights`

下载yolov4-tiny.weights的权重，也可以在浏览器下好放进darknet文件夹下

6. 输入 `./darknet detector demo cfg/coco.data cfg/yolov4-tiny.cfg yolov4-tiny.weights "nvarguscamerasrc ! video/x-raw(memory:NVMM), width=1280, height=720, format=NV12, framerate=30/1 ! nvvidconv flip-method=2 ! video/x-raw, width=1280, height=720, format=BGRx ! videoconvert ! video/x-raw, format=BGR ! appsink"`

即可使用darknet框架的yolov4-tiny推理CSI摄像头的视频，此方法参考 <https://blog.csdn.net/x16516581/article/details/100570038>这位博主，具体使用请看此帖



在不使用TensorRT加速，1280*720的视频，在416 416的输入下，yolov4-tiny约为12.7fps/s

五、torch & torchvision 安装 推理yolov5s

此处参考官方教程<https://forums.developer.nvidia.com/t/pytorch-for-jetson-version-1-9-0-now-available/72048>



1. 下载官方提供的torch-1.8.0-cp36-cp36m-linux_aarch64.whl包
下载地址: <https://nvidia.box.com/shared/static/p57jwntv436lfrd78inwl7iml6p13fzh.whl>
百度网盘下载链接:
链接:<https://pan.baidu.com/s/1TDNwzQIzP9B20NXLysIn0g>
提取码:7q73
2. `sudo apt-get install python3-pip libopenblas-base libopenmpi-dev`



3. `pip3 install Cython`



4. `pip3 install numpy torch-1.8.0-cp36-cp36m-linux_aarch64.whl` #注意路径下包含刚才下载的“torch-1.8.0”这个包
5. `sudo apt-get install libjpeg-dev zlib1g-dev libpython3-dev libavcodec-dev libavformat-dev libswscale-dev`



6. 注意torch torchvision版本对应



7. `git clone --branch v0.9.0 https://github.com/pytorch/vision torchvision`



8. `cd torchvision`

9. `export BUILD_VERSION=0.9.0`

10. `python3 setup.py install --user`



编译安装需要一段时间，请耐心等待！安装完成如图所示

接下来进入yolov5环境搭建



1. `git clone https://github.com/ultralytics/yolov5.git`
2. `python3 -m pip install --upgrade pip`



3. 在yolov5项目下，右键打开终端 `pip3 install -r requirements -i https://mirror.baidu.com/pypi/sample`



4. 等环境配置好后，`python3 detect.py` 第一次运行会下载yolov5s.pt的权重，下载完成，推理结果如下



48



12



登录bilibili, 享受更多权益!



免费看高清视频



多端同步播放记录



发表弹幕/评论



热门番剧影视看不停

立即登录

Jetson nano从烧录系统到DeepStream+TensorRT+yolov5检测C...



才疏学浅的...

+ 关注



六、TensorRT make & 推理测试

1. `git clone https://github.com/wang-xinyu/tensorrtx.git`



2. 具体编译方法可参考

<https://github.com/wang-xinyu/tensorrtx/blob/master/yolov5/README.md>



```
cd tensorrtx
cp yolov5/gen_wts.py ~/yolov5
cd ~/yolov5
python3 gen_wts.py yolov5s.pt
```



```
cd ~/tensorrtx/yolov5  
mkdir build && cd build  
mv ~/yolov5/yolov5s.wts ./  
cmake ..
```



```
make -j
```



```
./yolov5 -s yolov5s.wts yolov5s.engine s #生成engine文件需要一段时间，请耐心等待  
./yolov5 -d yolov5s.engine ../samples
```



推理结果如图所示，经过FP16 TensorRT加速后，速度快了不少

七、DeepStream安装 & yolov5检测CSI摄像头视频

1. DeepStream安装可参考官方教程

<https://developer.nvidia.com/deepstream-getting-started>

https://docs.nvidia.com/metropolis/deepstream/dev-guide/text/DS_Quickstart.html#jetson-setup



2. 下载deepstream_sdk_v5.1.0_jetson.tbz2

链接: <https://developer.nvidia.com/deepstream-sdk-v510-jetsontbz2>



Install Dependencies



这里选用 **Method 2**: Using the DeepStream tar package

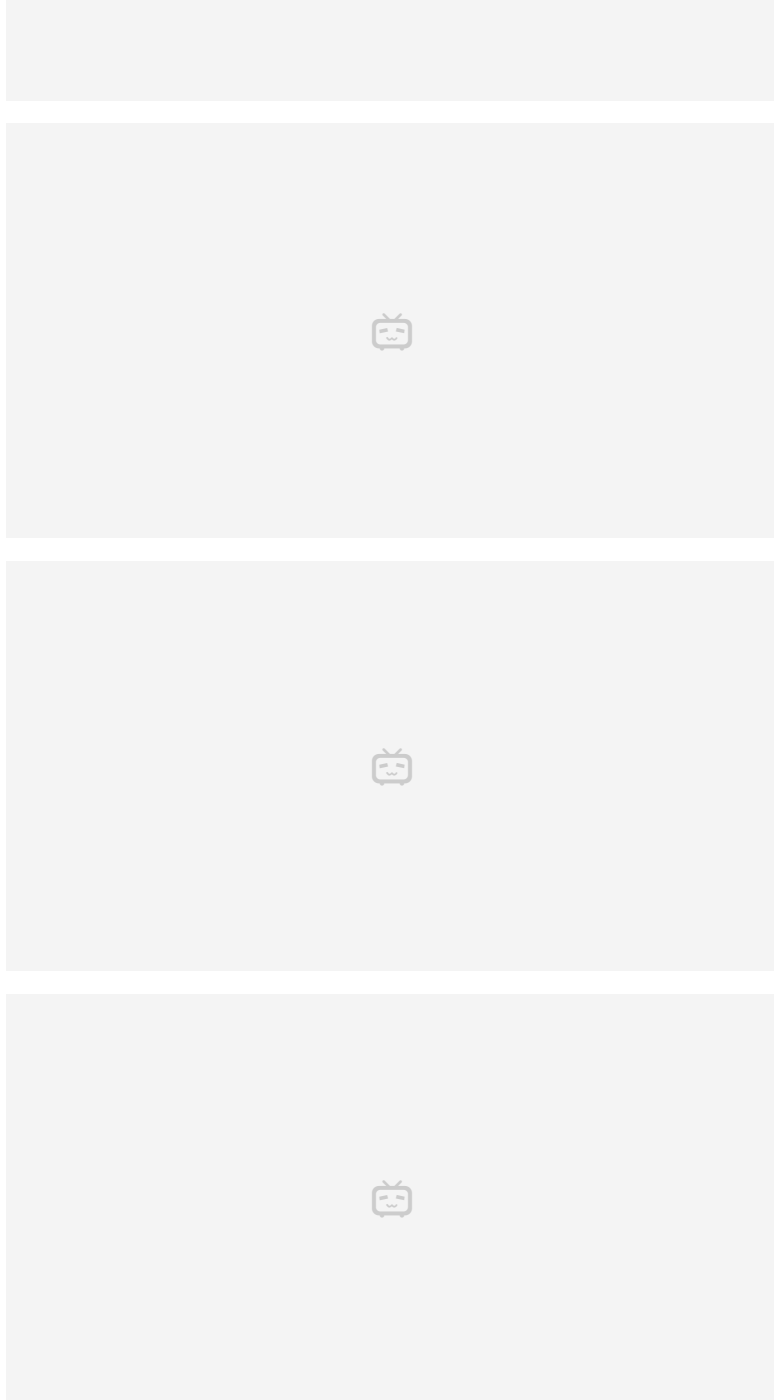


安装完成进入官方例程文件夹

```
cd /opt/nvidia/deepstream/deepstream-5.1/samples/configs/deepstream-app/
```

```
deepstream-app -c source8_1080p_dec_infer-resnet_tracker_tiled_display_fp16_nano.txt
```





效果如图所示，8路视频平均27.96 fps/s

yolov5检测

可以参考这两篇博客

<https://blog.csdn.net/zong596568821xp/article/details/109444343>

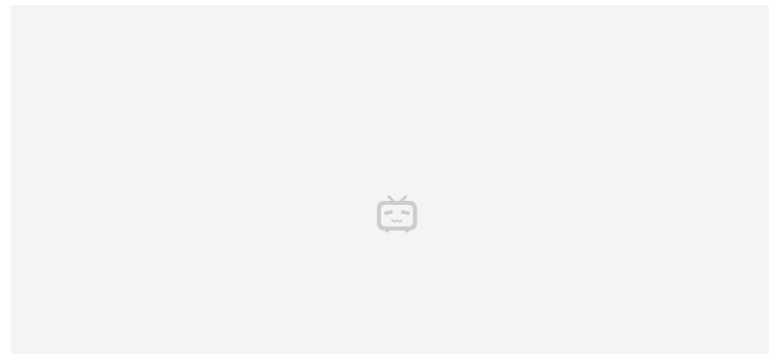
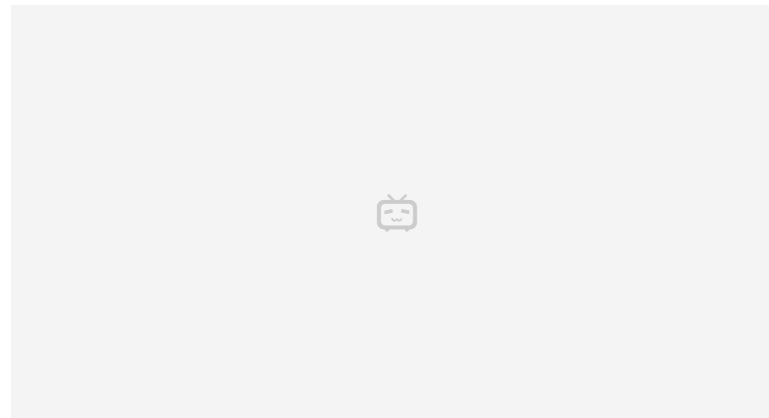
<https://blog.csdn.net/Tosonw/article/details/104154090>

1. `git clone https://github.com/DanaHan/Yolov5-in-Deepstream-5.0.git`

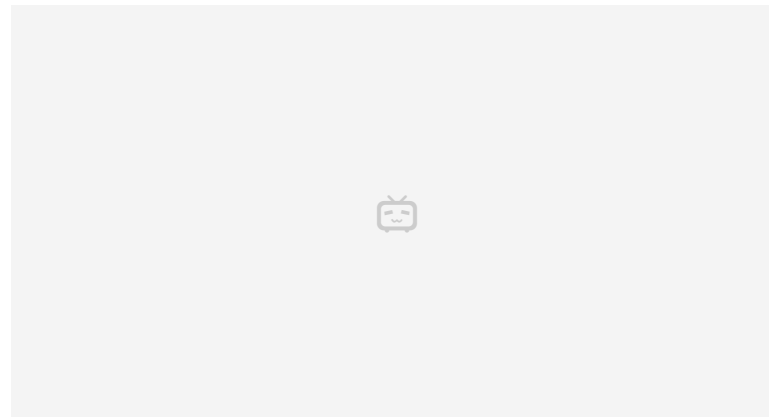


2. `cd Yolov5-in-Deepstream-5.0/Deepstream 5.0`

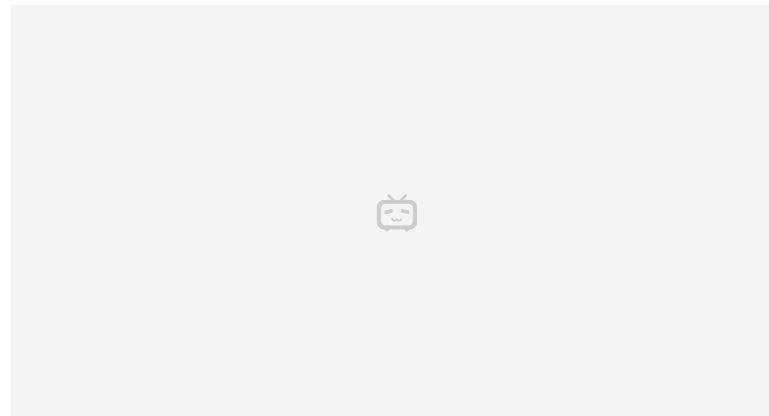
`cp ~/darknet/data/coco.names ./labels.txt`



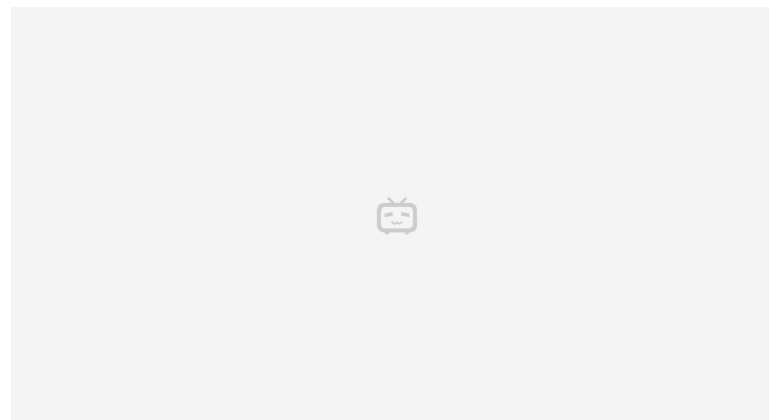
3. `vim config_infer_primary_yolov5.txt` #修改第59行和24行如图所示



4. `cp ~/tensorrtx/yolov5/build/yolov5s.engine ./` #复制之前生成的引擎文件到当前目录



5. `cd nvdsinfer_custom_impl_Yolo`

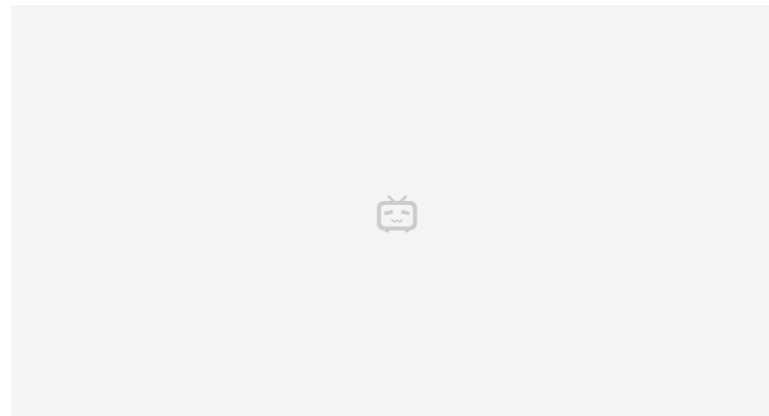


6. `make -j` # 生成libnvdnsinfer_custom_impl_Yolo.so文件

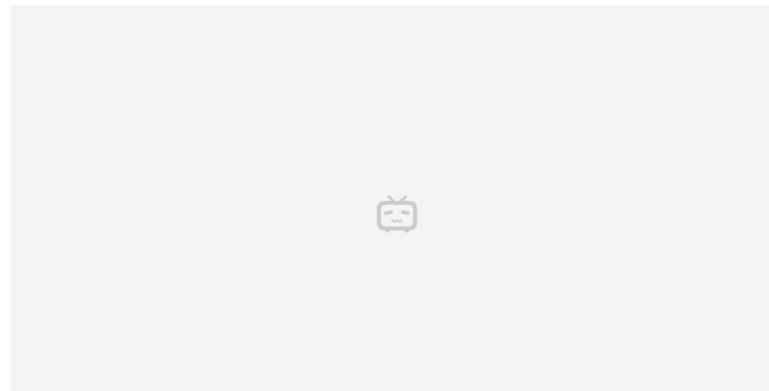


7. `cd ..` #返回到DeepStream5.0/下

`LD_PRELOAD=./libmyplugins.so deepstream-app -c deepstream_app_config_yoloV5.txt`



yolov5s 640输入 1080P的视频 帧率约12fps/s



8. 如图所示，修改config_infer_primary_yolov5.txt这个文件



LD_PRELOAD=./libmyplugins.so deepstream-app -c deepstream_app_config_yoloV5.txt

即可检测CSI摄像头画面！！ #因为手机屏幕放的距离摄像头有些近，所以画面有些模糊

本文为我原创 本文禁止转载或摘编

TensorRT Jetson nano yolov4 yolov5 DeepStream

分享到:



投诉或建议

推荐文章

更多精彩内容>

智汉科技推出蓝牙锂电池智能保护板解决方案

随着电动车的兴起，其带动了一整套产业链的勃兴，动力电池就是其中之一，很好的带动了电池技术的革新和扩产速度。而另...

RF Gray 疯狂的RF 学习 2 0 0



PMP|项目整合管理为何如此重要?

点击进入查看全文>

深圳弘德教育 学习 1 0 0



Java：使用 JPA 了解 Java 对持久性的支持

Java 平台已经有标准的 API 以 JDBC API 的形式与数据库系统一起工作。这些 API 非常擅长使用数据库，并提供了从 Java ...

粤嵌科技 学习 0 0 0



48



评论

全部评论

按时间排序



登录bilibili, 享受更多权益!



免费看高清视频

多端同步播放记录

发表弹幕/评论

热门番剧影视看不停

立即登录