

## 在Nvidia Jetson Nano上部署AI程序

笔记本: Docker

创建时间: 2021/2/3 10:27

更新时间: 2021/3/5 14:23

作者: Lee

URL: <https://ngc.nvidia.com/catalog/containers/nvidia:deepstream-l4t>

---

## 拉容器

```
docker pull nvcr.io/nvidia/deepstream-l4t:5.0.1-20.09-base
```

## 运行容器

要运行容器:

1.允许外部应用程序连接到主机的X显示器:

```
xhost +
```

2.使用nvidia-docker运行docker容器 (在下面的命令行中使用所需的容器标签):

```
sudo docker run -it --rm --net=host --runtime nvidia -e DISPLAY=$DISPLAY -w /opt/nvidia/deepstream/deepstream-5.0 -v /tmp/.X11-unix:/tmp/.X11-unix nvcr.io/nvidia/deepstream-l4t:5.0.1-20.09-samples
```

选项说明:

- -it表示以交互方式运行
- --rm将在完成后删除容器
- -v是挂载目录, 用于在容器文件系统中挂载主机的X11显示
- 5.0.1-20.09-samples是图像的标签; 20.09是指该发行版的容器版本; 样品是指容器的变体
- 用户可以根据需要挂载其他目录 (使用-v选项), 其中包含配置文件和模型, 以供从容器内执行的应用程序访问
- 另外, 需要包含--cap-add SYSLOG选项, 以启用容器内nvds\_logger的功能。

请参阅容器内的/opt/nvidia/deepstream/deepstream-5.0/README，以获取有关 deepstream-app使用情况的信息。要添加到上述docker命令以从Docker访问CSI摄像机的附加参数：-v / tmp / argus\_socket: / tmp / argus\_socket对于USB摄像机，附加参数--device / dev / video

# 安装环境

---

## 1.update

```
apt-get update
```

## 2.安装pip3

```
apt-get install python3-pip
```

## 3.安装PyQt5

```
apt-get install python3-pyqt5
```

## 4.安装gedit

```
apt-get install gedit
```

## 5.安装numpy

```
apt-get install python3-numpy
```

## 6.安装onnx

```
pip3 install onnx==1.4.1
```

## 7.安装PyCuda

- git clone

```
git clone https://github.com/jkjung-avt/tensorrt_demos.git
```

- 切换至ssd目录

```
cd ssd
```

- 安装pycuda

```
cat ./install_pycuda.sh  
./install_pycuda.sh
```

## 8.编译yolo plugins

```
cd tensorrt_demos  
cd plugins  
make
```

## 9.extra

```
sudo apt-get install python3-opencv  
  
sudo apt-get install python3-gi  
  
sudo apt install python3 python3-gst-1.0 \  
    gstreamer1.0-plugins-base \  
    gir1.2-gst-rtsp-server-1.0
```

## 10.GPIO

```
pip3 install Jetson.GPIO
```

激活GPIO

<https://blog.csdn.net/LK007CX/article/details/105384487>

# 保存当前docker实例

将已安装依赖环境的容器实例保存到磁盘。

commit

```
sudo docker commit 187acd017c8e gism-dev:v1.0
```

save

```
sudo docker save -o gism-v1.0.img gism-dev:v.1.0
```

## 编写Dockerfile

---

```
FROM gism:v1.0
WORKDIR /app
COPY . .
CMD ["/usr/bin/python3", "main.py"]
```

## build

---

```
sudo docker build -t gism:v2.0 .
```

## run docker container

---

```
xhost +
```

```
sudo docker run -it --net=host --privileged --runtime nvidia -e
DISPLAY=$DISPLAY gism:v2.0
```

删除docker image

```
docker image rm image_id...
```

---

extra:

docker image is being used by stoped container?

docker ps -a

docker rm container\_id

docker rmi image\_id

docker GPIO权限问题

参照edgeEye

--privileged

```
docker stop $(docker ps -aq)
xhost +
docker run \
--rm \
--privileged \
--net=host \
--runtime nvidia \
-e DISPLAY=$DISPLAY \
-v /tmp/.X11-unix:/tmp/.X11-unix \
-v /etc/localtime:/etc/localtime:ro \
-v $HOME/EdgeEye:/opt/EdgeEye tegra-cuda-base-run:v5
```

微软雅黑字体;

<https://www.cnblogs.com/weifeng1463/p/10475713.html>

docker中文乱码

step1

[pianshen.com/article/64521449753](https://pianshen.com/article/64521449753)

step2

<https://segmentfault.com/a/1190000005026503>

**容器持久化**

新建一个卷

```
docker volume create appconfig
```

每次使用docker都要输入密码

解决docker每次都需要输入sudo的权限问题

每次在使用docker命令的时候，都要添加sudo权限，否则就会报

下面我们来添加权限

1、输入sudo groupadd docker

回显groupadd: group 'docker' already exists

2、将当前用户添加到docker用户组

sudo gpasswd -a \$(whoami) docker

或者添加指定的用户sudo gpasswd -a <用户名> docker

例如：sudo gpasswd -a ubuntu docker

3、重启docker服务

sudo service docker restart #ubuntu

sudo systemctl restart docker.service #Centos

