

Jetson Orin Nano 安裝與測試 YOLOv5 教學講義

一、前置環境：

- 系統：Ubuntu 20.04 / 22.04 (JetPack 5.x / 6.x)
- Python 3.8+ , CUDA 11.x , cuDNN、TensorRT 皆已安裝

二、安裝 YOLOv5：

```
sudo apt update
```

```
sudo apt install -y git python3-pip python3-venv libopencv-dev
```

```
python3 -m venv yolov5-env && source yolov5-env/bin/activate
```

```
git clone https://github.com/ultralytics/yolov5.git && cd yolov5
```

```
pip install torch==1.13.0+nv22.10 torchvision==0.14.0+nv22.10 -f https://developer.download.nvidia.com/compute/redist/jc592363
```

```
pip install -r requirements.txt
```

三、執行圖片推論：

```
python detect.py --weights yolov5s.pt --source data/images/bus.jpg --device 0
```

四、執行 USB 攝影機推論：

```
python detect.py --weights yolov5s.pt --source 0 --device 0
```

五、常見排錯：

- OpenCV libGL.so 缺失：sudo apt install libgl1
- 執行速度慢：改用 yolov5n.pt，或降低解析度

六、加速部署（選用）：

```
python export.py --weights yolov5s.pt --include onnx
```

```
/usr/src/tensorrt/bin/trtexec --onnx=yolov5s.onnx --saveEngine=yolov5s.engine --fp16
```

七、驗證 GPU 是否啟用：

```
import torch
```

```
print(torch.cuda.is_available())
```

```
print(torch.cuda.get_device_name(0))
```

八、成功標準：

- 可正常執行 detect.py 並偵測物體
- 結果保存在 runs/detect/exp
- 執行過程 GPU 有參與 (jtop 或 tegrastats)