Yolo model optimizations on OrangePi

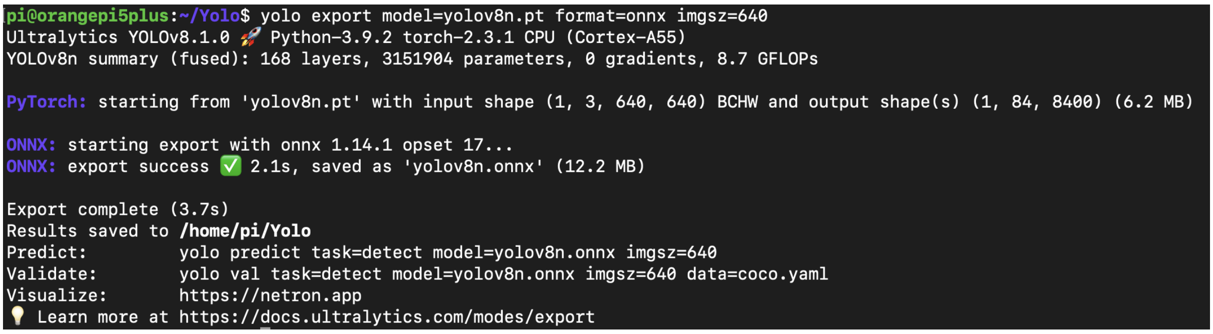
◷ Created	@August 3, 2024 11:40 AM
⊙ Class	Embedded AI

Homework 3 Report

.pt 2 onnx

This conversion usually works out of the box as it did this time:

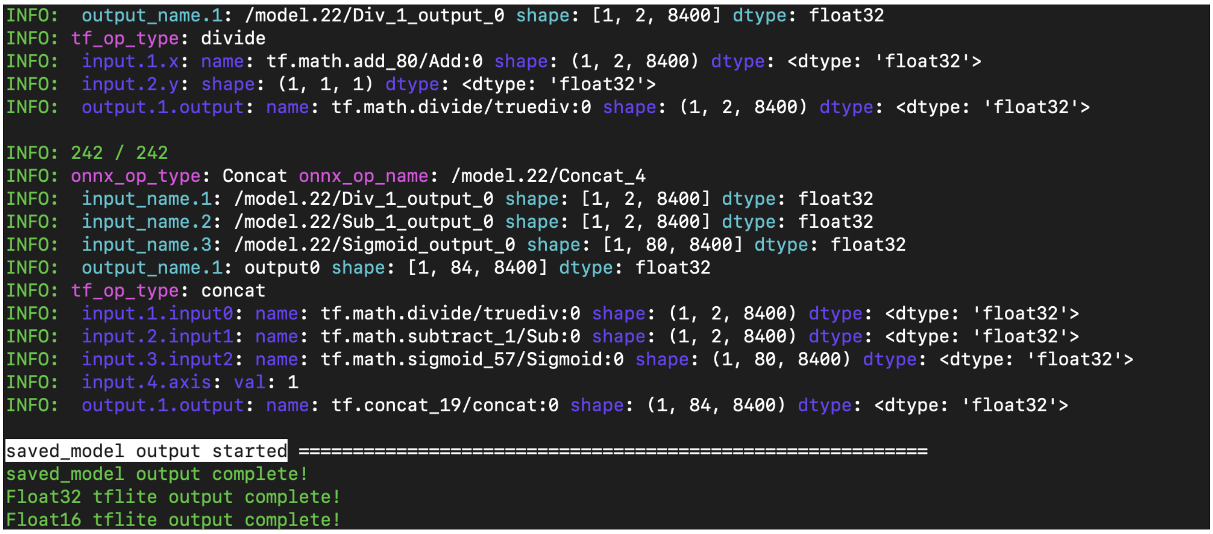
```
yolo export model=yolov8n.pt format=onnx imgsz=640
```



```
pi@orangepi5plus:~/Yolo$ yolo export model=yolov8n.pt format=onnx imgsz=640
Ultralytics YOLOv8.1.0 🚀 Python-3.9.2 torch-2.3.1 CPU (Cortex-A55)
YOLOv8n summary (fused): 168 layers, 3151904 parameters, 0 gradients, 8.7 GFLOPs

PyTorch: starting from 'yolov8n.pt' with input shape (1, 3, 640, 640) BCHW and output shape(s) (1, 84, 8400) (6.2 MB)

ONNX: starting export with onnx 1.14.1 opset 17...
ONNX: export success ✅ 2.1s, saved as 'yolov8n.onnx' (12.2 MB)

Export complete (3.7s)
Results saved to /home/pi/Yolo
Predict:         yolo predict task=detect model=yolov8n.onnx imgsz=640
Validate:        yolo val task=detect model=yolov8n.onnx imgsz=640 data=coco.yaml
Visualize:       https://netron.app
💡 Learn more at https://docs.ultralytics.com/modes/export
```

Onnx 2 TFLite

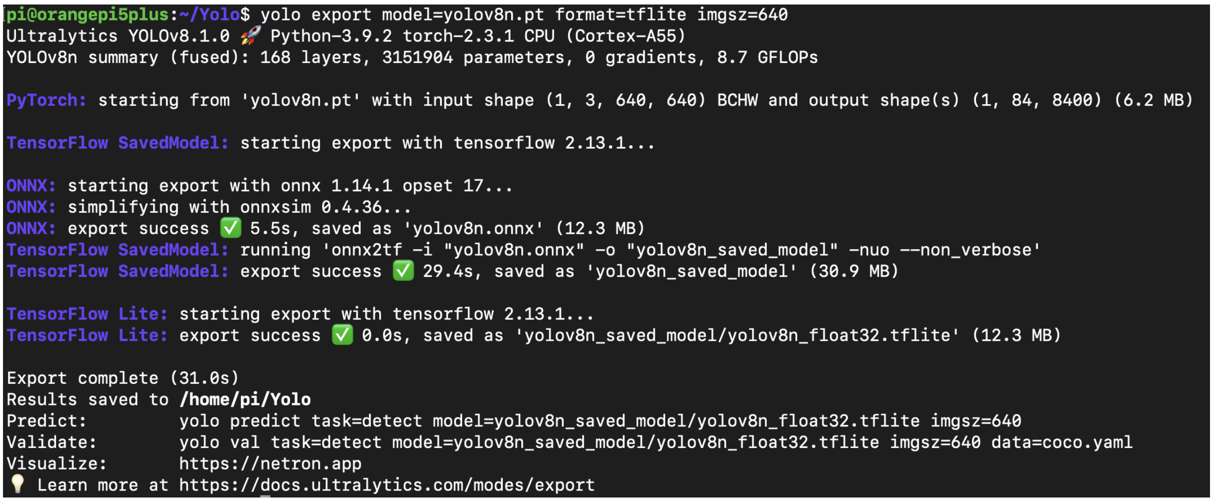
```
onnx2tf -i "yolov8n.onnx" -o "yolov8n_tflite" -nuo
```



```
INFO:  output_name.1: /model.22/Div_1_output_0 shape: [1, 2, 8400] dtype: float32
INFO: tf_op_type: divide
INFO:  input.1.x: name: tf.math.add_80/Add:0 shape: (1, 2, 8400) dtype: <dtype: 'float32'>
INFO:  input.2.y: shape: (1, 1, 1) dtype: <dtype: 'float32'>
INFO:  output.1.output: name: tf.math.divide/truediv:0 shape: (1, 2, 8400) dtype: <dtype: 'float32'>

INFO: 242 / 242
INFO: onnx_op_type: Concat onnx_op_name: /model.22/Concat_4
INFO:  input_name.1: /model.22/Div_1_output_0 shape: [1, 2, 8400] dtype: float32
INFO:  input_name.2: /model.22/Sub_1_output_0 shape: [1, 2, 8400] dtype: float32
INFO:  input_name.3: /model.22/Sigmoid_output_0 shape: [1, 80, 8400] dtype: float32
INFO:  output_name.1: output0 shape: [1, 84, 8400] dtype: float32
INFO: tf_op_type: concat
INFO:  input.1.input0: name: tf.math.divide/truediv:0 shape: (1, 2, 8400) dtype: <dtype: 'float32'>
INFO:  input.2.input1: name: tf.math.subtract_1/Sub:0 shape: (1, 2, 8400) dtype: <dtype: 'float32'>
INFO:  input.3.input2: name: tf.math.sigmoid_57/Sigmoid:0 shape: (1, 80, 8400) dtype: <dtype: 'float32'>
INFO:  input.4.axis: val: 1
INFO:  output.1.output: name: tf.concat_19/concat:0 shape: (1, 84, 8400) dtype: <dtype: 'float32'>

saved_model output started ==========================================================
saved_model output complete!
Float32 tflite output complete!
Float16 tflite output complete!
```

.pt 2 TFLite (More complex method dependency-wise)

```
yolo export model=yolov8n.pt format=tflite imgsz=640
```



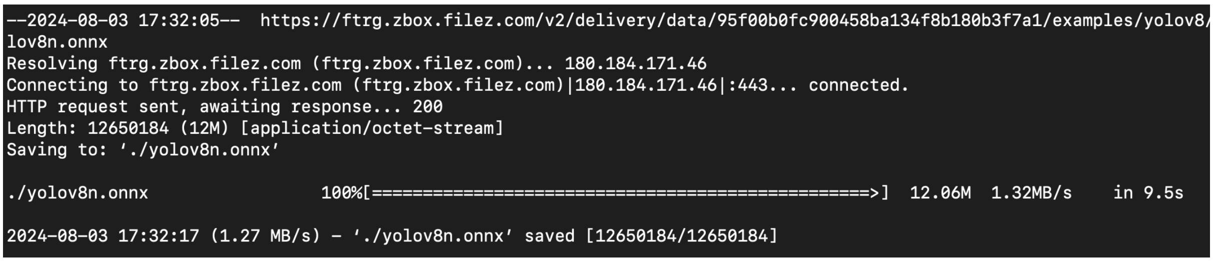
```
pi@orangepi5plus:~/Yolo$ yolo export model=yolov8n.pt format=tflite imgsz=640
Ultralytics YOLOv8.1.0 🚀 Python-3.9.2 torch-2.3.1 CPU (Cortex-A55)
YOLOv8n summary (fused): 168 layers, 3151904 parameters, 0 gradients, 8.7 GFLOPs

PyTorch: starting from 'yolov8n.pt' with input shape (1, 3, 640, 640) BCHW and output shape(s) (1, 84, 8400) (6.2 MB)

TensorFlow SavedModel: starting export with tensorflow 2.13.1...

ONNX: starting export with onnx 1.14.1 opset 17...
ONNX: simplifying with onnxsim 0.4.36...
ONNX: export success ✅ 5.5s, saved as 'yolov8n.onnx' (12.3 MB)
TensorFlow SavedModel: running 'onnx2tf -i "yolov8n.onnx" -o "yolov8n_saved_model" -nuo --non_verbose'
TensorFlow SavedModel: export success ✅ 29.4s, saved as 'yolov8n_saved_model' (30.9 MB)

TensorFlow Lite: starting export with tensorflow 2.13.1...
TensorFlow Lite: export success ✅ 0.0s, saved as 'yolov8n_saved_model/yolov8n_float32.tflite' (12.3 MB)

Export complete (31.0s)
Results saved to /home/pi/Yolo
Predict:         yolo predict task=detect model=yolov8n_saved_model/yolov8n_float32.tflite imgsz=640
Validate:        yolo val task=detect model=yolov8n_saved_model/yolov8n_float32.tflite imgsz=640 data=coco.yaml
Visualize:       https://netron.app
💡 Learn more at https://docs.ultralytics.com/modes/export
```

In order to make this work I had to downgrade tflite (The precompiled Tensorflow package wants a newer libstdc++ than is provided with Bullseye)

```
python3 -m pip install --upgrade tflite-support==0.4.2
python3 -m pip install --upgrade tflite-runtime==2.11.0
```

Onnx to RKNN on RK3588

```
git clone https://github.com/airockchip/rknn_model_zoo
cd rknn_model_zoo/examples/yolov8/model
bash download_model.sh

cd ../python/
git clone https://github.com/airockchip/rknn-toolkit2/
```



```
--2024-08-03 17:32:05--  https://ftrg.zbox.filez.com/v2/delivery/data/95f00b0fc900458ba134f8b180b3f7a1/examples/yolov8/
lov8n.onnx
Resolving ftrg.zbox.filez.com (ftrg.zbox.filez.com)... 180.184.171.46
Connecting to ftrg.zbox.filez.com (ftrg.zbox.filez.com)|180.184.171.46|:443... connected.
HTTP request sent, awaiting response... 200
Length: 12650184 (12M) [application/octet-stream]
Saving to: ‘./yolov8n.onnx’

./yolov8n.onnx                 100%[=================================================>]  12.06M  1.32MB/s    in 9.5s

2024-08-03 17:32:17 (1.27 MB/s) – ‘./yolov8n.onnx’ saved [12650184/12650184]
```

Inference speed comparison

Model	Average Inference Time, ms	Average FPS
yolov8n_float32.tflite	520	1,92
yolov8n_float16.tflite	500	2
yolov8n.pt	410	2,4
yolov8n.onnx	275	3,6
yolov8n.rknn	42	24