Tina V853 NPU Yolact 模型部署实战

版本号: 1.0
发布日期: 2021.07.21

版本历史

版本号	日期	制/修订人	内容描述
1.0	2021.07.21	PDC	NPU Yolact 模型部署实战

目　录

插　图

1 前言

1.1 读者对象

本文档（本指南）主要适用于以下人员：

- 技术支持工程师
- 软件开发工程师
- AI 应用案客户

1.2 约定

1.2.1 符号约定

本文中可能出现的符号如下：

⚠ **警告**

警告

💡 技巧

1. 技巧
2. 小常识

📖 说明

说明

2 正文

2.1 NPU 开发简介

- 支持 int8/uint8/int16 量化精度，运算性能可达 1TOPS.
- 相较于 GPU 作为 AI 运算单元的大型芯片方案，功耗不到 GPU 所需要的 1%.
- 可直接导入 Caffe, TensorFlow, Onnx, TFLite，Keras, Darknet, pyTorch 等模型格式.
- 提供 AI 开发工具：支持模型快速转换、支持开发板端侧转换 API、支持 TensorFlow, TF Lite, Caffe, ONNX, Darknet, pyTorch 等模型.
- 提供 AI 应用开发接口：提供 NPU 跨平台 API.

2.2 开发流程

NPU 开发完整的流程如下图所示：

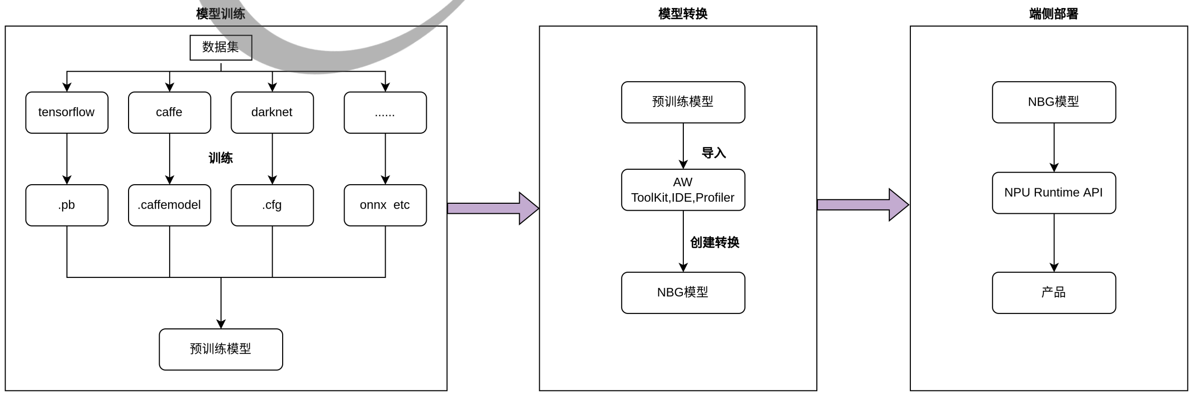


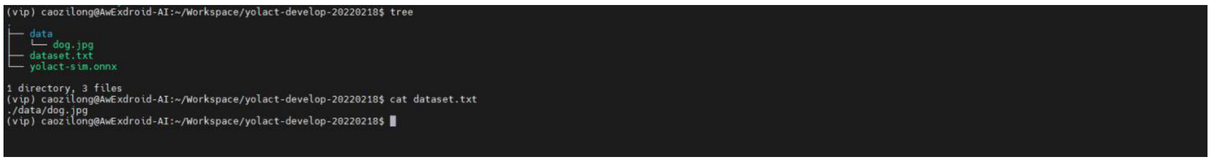
图 2-1: npu_1.png

2.3 获取 YOLACT 原始模型

YOLACT 模型获取方式有很多种，可以基于项目 https://github.com/dbolya/yolact 产生 onnx 格式的 yolact 模型，本文档假设你已经产生了 yolact.onnx 模型

2.4 模型部署工作目录结构

yolact-sim.onnx 有 120M，还是蛮大的.



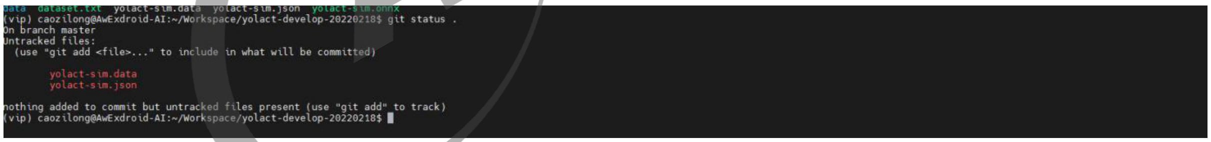
```
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ tree
.
├── data
│   └── dog.jpg
├── dataset.txt
└── yolact-sim.onnx

1 directory, 3 files
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ cat dataset.txt
./data/dog.jpg
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
```

图 2-2: npu_workspace

2.5 导入模型

```
pegasus import onnx --model yolact-sim.onnx --output-model yolact-sim.json --output-data yolact-sim
.data
```

导入模型的目的是将开放模型转换为符合 VIP 模型网络描述文件 (.json) 和权重文件 (.data)



```
data  dataset.txt  yolact-sim.data  yolact-sim.json  yolact-sim.onnx
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ git status .
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        yolact-sim.data
        yolact-sim.json

nothing added to commit but untracked files present (use "git add" to track)
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
```

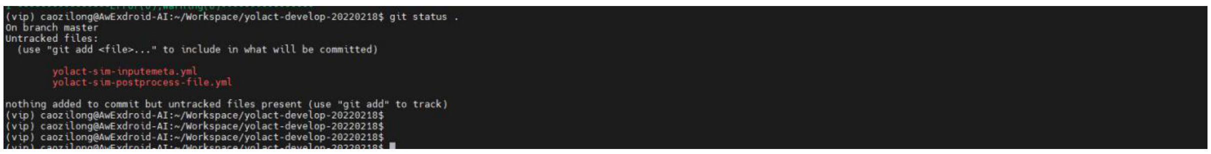
图 2-3: npu_import

2.6 创建 YML 文件

YML 文件对网络的输入和输出进行描述，比如输入图像的形状，归一化系数 (均值，零点)，图像格式，输出 tensor 的输出格式，后处理方式等等，命令如下：

```
pegasus generate inputmeta --model yolact-sim.json --input-meta-output yolact-sim-inputemeta.yml
```

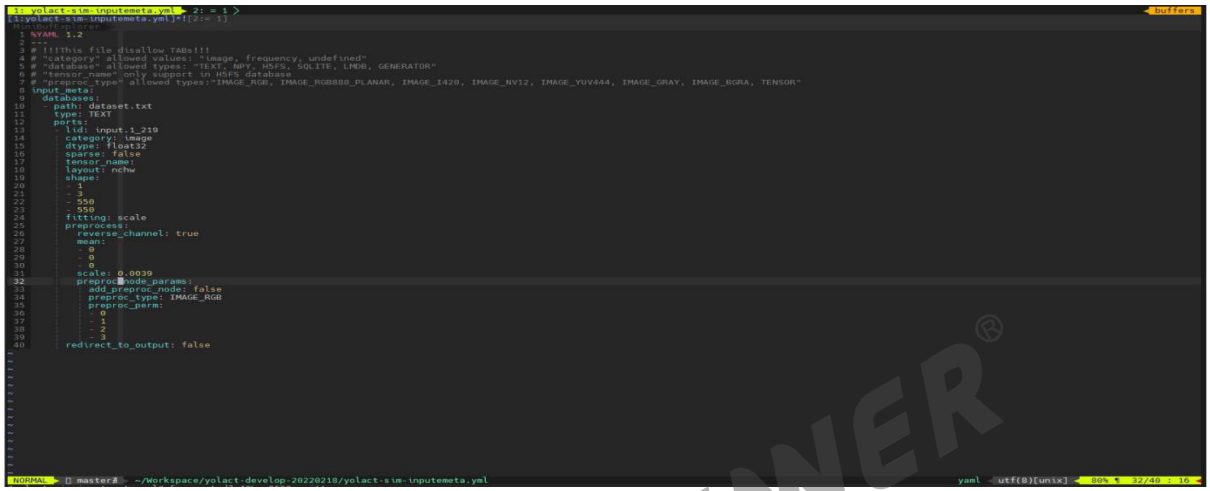
```
pegasus generate postprocess-file --model yolact-sim.json --postprocess-file-output yolact-sim-
postprocess-file.yml
```



```
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ git status .
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        yolact-sim-inputemeta.yml
        yolact-sim-postprocess-file.yml

nothing added to commit but untracked files present (use "git add" to track)
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
```

图 2-4: npu_yml

修改 input meta 文件中的的 scale 参数为 0.0039(1/256)，和 yolov3 的改法完全一致。



```
 1: yolact-sim-inputemeta.yml  2: = 1 >
[1:yolact-sim-inputemeta.yml]*![2:= 1]
%YAML 1.2
---
# !!!This file disallow TABs!!!
# "category" allowed values: "image, frequency, undefined"
# "database" allowed types: "TEXT, NPY, H5FS, SQLITE, LMDB, GENERATOR"
# "tensor_name" only support in H5FS database
# "preproc_type" allowed types:"IMAGE_RGB, IMAGE_RGB888_PLANAR, IMAGE_I420, IMAGE_NV12, IMAGE_YUV444, IMAGE_GRAY, IMAGE_BGRA, TENSOR"
input_meta:
  databases:
  - path: dataset.txt
    type: TEXT
    ports:
    - lid: input.1_219
      category: image
      dtype: float32
      sparse: false
      tensor_name:
      layout: nchw
      shape:
      - 1
      - 3
      - 550
      - 550
      fitting: scale
      preprocess:
        reverse_channel: true
        mean:
        - 0
        - 0
        - 0
        scale: 0.0039
        preproc_node_params:
          add_preproc_node: false
          preproc_type: IMAGE_RGB
          preproc_perm:
          - 0
          - 1
          - 2
          - 3
      redirect_to_output: false
NORMAL  master#  ~/Workspace/yolact-develop-20220218/yolact-sim-inputemeta.yml    yaml  utf(8)[unix]  80%  32/40 : 16
```

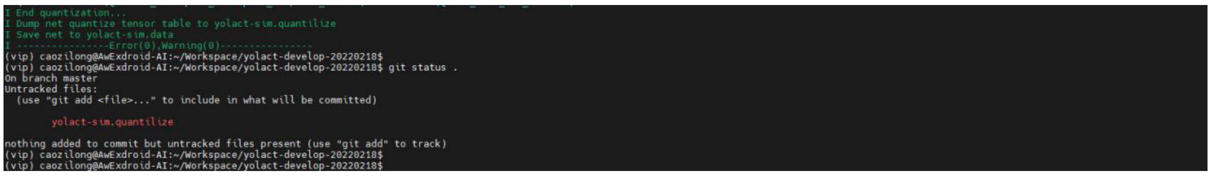
图 2-5: npu_scale

2.7 量化

量化命令：

```
pegasus quantize --model yolact-sim.json  --model-data yolact-sim.data --batch-size 1 --device CPU
 --with-input-meta yolact-sim-inputemeta.yml --rebuild --model-quantize yolact-sim.quantilize --
quantizer asymmetric_affine --qtype uint8
```

命令执行后，创建了量化表文件

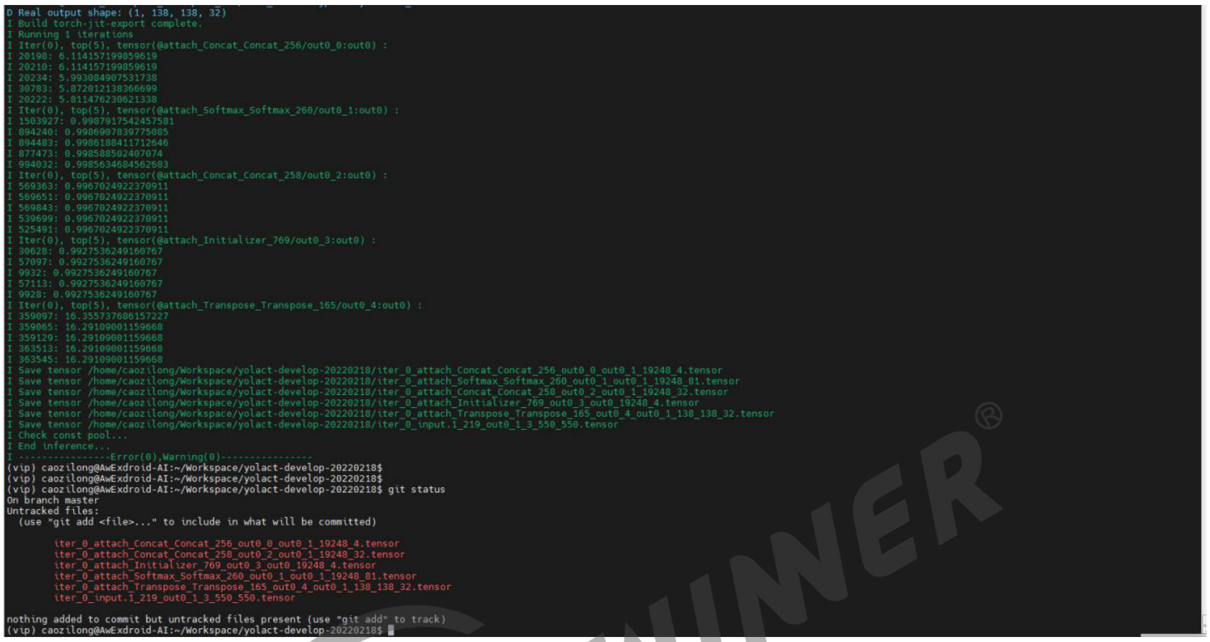


```
I End quantization...
I Dump net quantize tensor table to yolact-sim.quantilize
I Save net to yolact-sim.data
I ----------------Error(0),Warning(0)----------------
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ git status .
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        yolact-sim.quantilize

nothing added to commit but untracked files present (use "git add" to track)
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
```

图 2-6: npu_quantilize

2.8 预推理

```
pegasus inference --model yolact-sim.json --model-data yolact-sim.data --batch-size 1 --dtype
 quantized --model-quantize yolact-sim.quantilize --device CPU --with-input-meta yolact-sim-
inputemeta.yml --postprocess-file yolact-sim-postprocess-file.yml
```



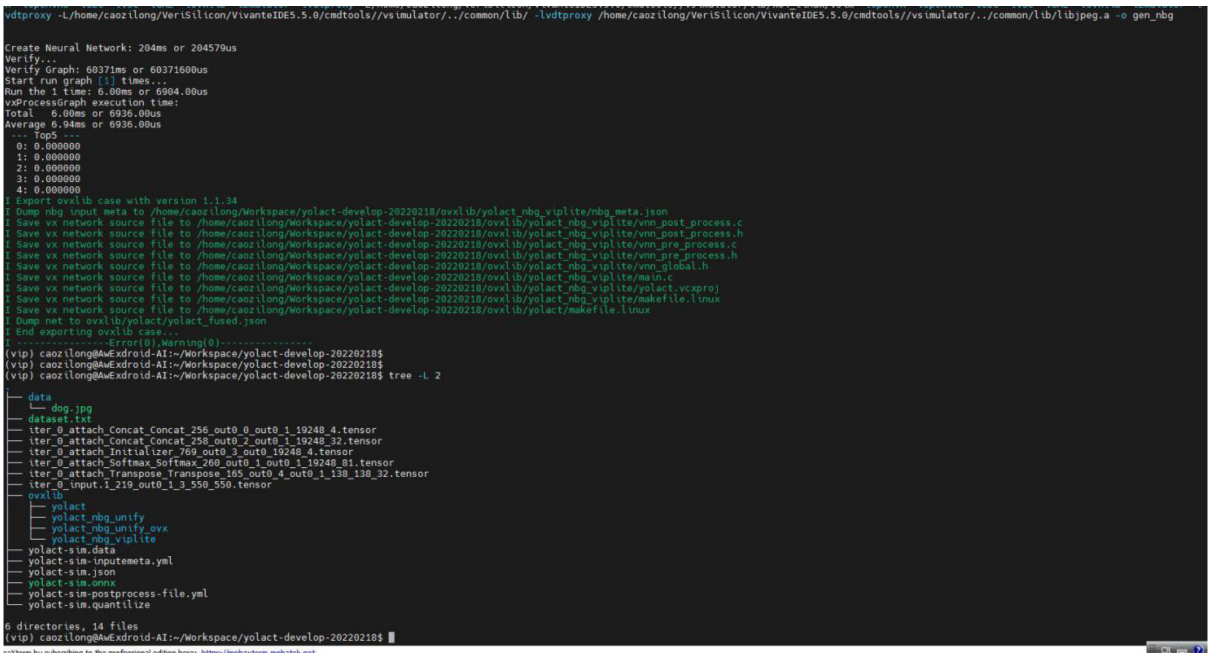
```
D Real output shape: (1, 138, 138, 32)
I Build torch-jit-export complete.
I Running 1 iterations
I Iter(0), top(5), tensor(@attach_Concat_Concat_256/out0_0:out0) :
I 20198: 6.114157199859619
I 20210: 6.114157199859619
I 20234: 5.993084907531738
I 30783: 5.872012138366699
I 20222: 5.811476230621338
I Iter(0), top(5), tensor(@attach_Softmax_Softmax_260/out0_1:out0) :
I 1503927: 0.9987917542457581
I 894240: 0.9986907839775085
I 894483: 0.9986188411712646
I 877473: 0.998588502407074
I 994032: 0.9985634684562683
I Iter(0), top(5), tensor(@attach_Concat_Concat_258/out0_2:out0) :
I 569363: 0.9967024922370911
I 569651: 0.9967024922370911
I 569843: 0.9967024922370911
I 539699: 0.9967024922370911
I 525491: 0.9967024922370911
I Iter(0), top(5), tensor(@attach_Initializer_769/out0_3:out0) :
I 30628: 0.9927536249160767
I 57097: 0.9927536249160767
I 9932: 0.9927536249160767
I 57113: 0.9927536249160767
I 9928: 0.9927536249160767
I Iter(0), top(5), tensor(@attach_Transpose_Transpose_165/out0_4:out0) :
I 359097: 16.355737686157227
I 359065: 16.29109001159668
I 359129: 16.29109001159668
I 363513: 16.29109001159668
I 363545: 16.29109001159668
I Save tensor /home/caozilong/Workspace/yolact-develop-20220218/iter_0_attach_Concat_Concat_256_out0_0_out0_1_19248_4.tensor
I Save tensor /home/caozilong/Workspace/yolact-develop-20220218/iter_0_attach_Softmax_Softmax_260_out0_1_out0_1_19248_81.tensor
I Save tensor /home/caozilong/Workspace/yolact-develop-20220218/iter_0_attach_Concat_Concat_258_out0_2_out0_1_19248_32.tensor
I Save tensor /home/caozilong/Workspace/yolact-develop-20220218/iter_0_attach_Initializer_769_out0_3_out0_19248_4.tensor
I Save tensor /home/caozilong/Workspace/yolact-develop-20220218/iter_0_attach_Transpose_Transpose_165_out0_4_out0_1_138_138_32.tensor
I Save tensor /home/caozilong/Workspace/yolact-develop-20220218/iter_0_input.1_219_out0_1_3_550_550.tensor
I Check const pool...
I End inference...
I ----------------Error(0),Warning(0)----------------
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        iter_0_attach_Concat_Concat_256_out0_0_out0_1_19248_4.tensor
        iter_0_attach_Concat_Concat_258_out0_2_out0_1_19248_32.tensor
        iter_0_attach_Initializer_769_out0_3_out0_19248_4.tensor
        iter_0_attach_Softmax_Softmax_260_out0_1_out0_1_19248_81.tensor
        iter_0_attach_Transpose_Transpose_165_out0_4_out0_1_138_138_32.tensor
        iter_0_input.1_219_out0_1_3_550_550.tensor

nothing added to commit but untracked files present (use "git add" to track)
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
```

图 2-7: npu_inf

2.9 导出代码和 NBG 文件

```
pegasus export ovxlib --model yolact-sim.json --model-data yolact-sim.data --dtype quantized --
model-quantize yolact-sim.quantilize --batch-size 1 --save-fused-graph --target-ide-project '
linux64' --with-input-meta yolact-sim-inputemeta.yml --postprocess-file yolact-sim-postprocess-file
.yml --output-path ovxlib/yolact/yolact --pack-nbg-unify  --optimize "VIP9000PICO_PID0XEE" --viv-
sdk ${VIV_SDK}
```

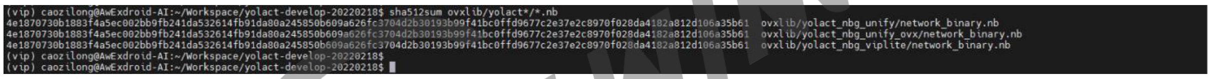
```
pegasus export ovxlib --model yolact-sim.json --model-data yolact-sim.data --dtype quantized --
model-quantize yolact-sim.quantilize --batch-size 1 --save-fused-graph --target-ide-project '
linux64' --with-input-meta yolact-sim-inputemeta.yml --postprocess-file yolact-sim-postprocess-file
.yml --output-path ovxlib/yolact/yolact --pack-nbg-viplite --optimize "VIP9000PICO_PID0XEE" --viv-
sdk ${VIV_SDK}
```



```
vdtproxy -L/home/caozilong/VeriSilicon/VivanteIDE5.5.0/cmdtools//vsimulator/../common/lib/ -lvdtproxy /home/caozilong/VeriSilicon/VivanteIDE5.5.0/cmdtools//vsimulator/../common/lib/libjpeg.a -o gen_nbg


Create Neural Network: 204ms or 204579us
Verify...
Verify Graph: 60371ms or 60371600us
Start run graph [1] times...
Run the 1 time: 6.00ms or 6904.00us
vxProcessGraph execution time:
Total   6.00ms or 6936.00us
Average 6.94ms or 6936.00us
 --- Top5 ---
  0: 0.000000
  1: 0.000000
  2: 0.000000
  3: 0.000000
  4: 0.000000
I Export ovxlib case with version 1.1.34
I Dump nbg input meta to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/nbg_meta.json
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/vnn_post_process.c
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/vnn_post_process.h
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/vnn_pre_process.c
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/vnn_pre_process.h
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/vnn_global.h
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/main.c
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/yolact.vcxproj
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact_nbg_viplite/makefile.linux
I Save vx network source file to /home/caozilong/Workspace/yolact-develop-20220218/ovxlib/yolact/makefile.linux
I Dump net to ovxlib/yolact/yolact_fused.json
I End exporting ovxlib case...
I ----------------Error(0),Warning(0)----------------
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ tree -L 2
.
├── data
│   └── dog.jpg
├── dataset.txt
├── iter_0_attach_Concat_Concat_256_out0_0_out0_1_19248_4.tensor
├── iter_0_attach_Concat_Concat_258_out0_2_out0_1_19248_32.tensor
├── iter_0_attach_Initializer_769_out0_3_out0_19248_4.tensor
├── iter_0_attach_Softmax_Softmax_260_out0_1_out0_1_19248_81.tensor
├── iter_0_attach_Transpose_Transpose_165_out0_4_out0_1_138_138_32.tensor
├── iter_0_input.1_219_out0_1_3_550_550.tensor
├── ovxlib
│   ├── yolact
│   ├── yolact_nbg_unify
│   ├── yolact_nbg_unify_ovx
│   └── yolact_nbg_viplite
├── yolact-sim.data
├── yolact-sim-inputemeta.yml
├── yolact-sim.json
├── yolact-sim.onnx
├── yolact-sim-postprocess-file.yml
└── yolact-sim.quantilize

6 directories, 14 files
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ █
```

图 2-8: npu_export

生成的 NBG 文件, 可以部署到端侧:

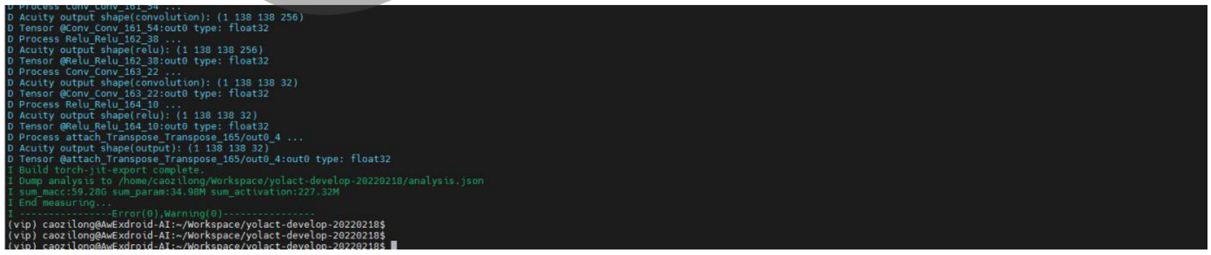


```
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ sha512sum ovxlib/yolact*/*.nb
4e1870730b1883f4a5ec002bb9fb241da532614fb91da80a245850b609a626fc3704d2b30193b99f41bc0ffd9677c2e37e2c8970f028da4182a812d106a35b61  ovxlib/yolact_nbg_unify/network_binary.nb
4e1870730b1883f4a5ec002bb9fb241da532614fb91da80a245850b609a626fc3704d2b30193b99f41bc0ffd9677c2e37e2c8970f028da4182a812d106a35b61  ovxlib/yolact_nbg_unify_ovx/network_binary.nb
4e1870730b1883f4a5ec002bb9fb241da532614fb91da80a245850b609a626fc3704d2b30193b99f41bc0ffd9677c2e37e2c8970f028da4182a812d106a35b61  ovxlib/yolact_nbg_viplite/network_binary.nb
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ █
```

图 2-9: npu_nbg

2.10 模型算力测量

```
pegasus measure --model yolact-sim.json
```



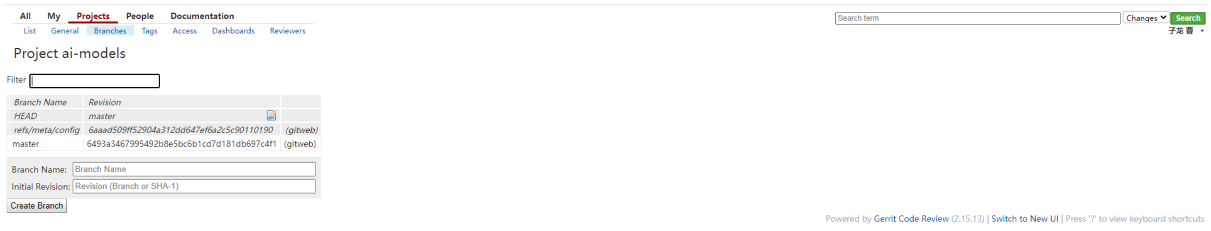
```
D Acuity output shape(convolution): (1 138 138 256)
D Tensor @Conv_Conv_161_54:out0 type: float32
D Process Relu_Relu_162_38 ...
D Acuity output shape(relu): (1 138 138 256)
D Tensor @Relu_Relu_162_38:out0 type: float32
D Process Conv_Conv_163_22 ...
D Acuity output shape(convolution): (1 138 138 32)
D Tensor @Conv_Conv_163_22:out0 type: float32
D Process Relu_Relu_164_10 ...
D Acuity output shape(relu): (1 138 138 32)
D Tensor @Relu_Relu_164_10:out0 type: float32
D Process attach_Transpose_Transpose_165/out0_4 ...
D Acuity output shape(output): (1 138 138 32)
D Tensor @attach_Transpose_Transpose_165/out0_4:out0 type: float32
I Build torch-jit-export complete.
I Dump analysis to /home/caozilong/Workspace/yolact-develop-20220218/analysis.json
I sum_macc:59.28G sum_param:34.98M sum_activation:227.32M
I End measuring...
I ----------------Error(0),Warning(0)----------------
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$
(vip) caozilong@AwExdroid-AI:~/Workspace/yolact-develop-20220218$ █
```

图 2-10: npu_measure

对比 yolov3 的 32.99G MACC 算力和 yolov3-tiny 的 2.79G macc 的算力需求，YOLACT 明显对算力的需求大很多. 题外话,1TOPS=1000GOPS, 对于 yolov3-tiny 来说，我们以 3G 的算力需求来计算，就是 0.003T，在 500M 频率下，NPU 的理论算力是 0.5T, 所以，YOLOV3 的帧率为：0.5T/0.003T=166 帧。这个值和仿真值是比较接近的.

2.11 后处理验证

yolact 的后处理 C 代码在如下位置：



Project ai-models

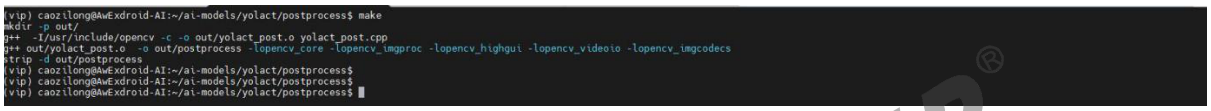
Filter

Branch Name	Revision	
HEAD	master	
refs/meta/config	6aaad509ff52904a312dd647ef6a2c5c90110190	(gitweb)
master	6493a3467995492b8e5bc6b1cd7d181db697c4f1	(gitweb)

Branch Name: Branch Name
Initial Revision: Revision (Branch or SHA-1)

图 2-11：npu_gerit

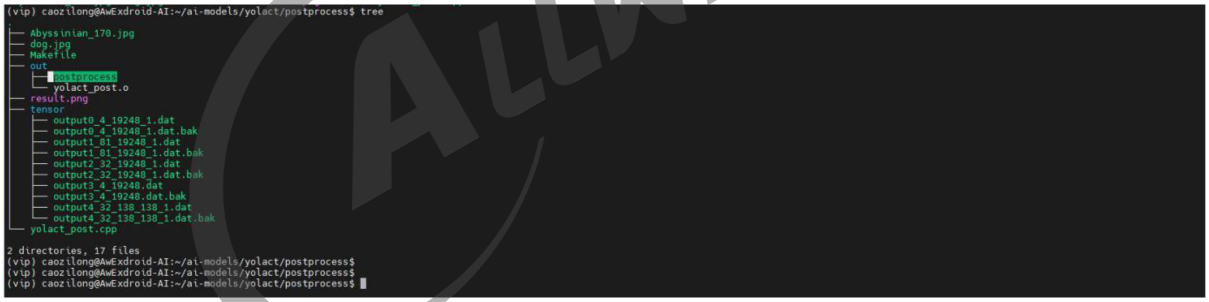
编译后处理模型：



```
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$ make
mkdir -p out/
g++  -I/usr/include/opencv -c -o out/yolact_post.o yolact_post.cpp
g++ out/yolact_post.o  -o out/postprocess -lopencv_core -lopencv_imgproc -lopencv_highgui -lopencv_videoio -lopencv_imgcodecs
strip -d out/postprocess
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$
```

图 2-12：npu_post

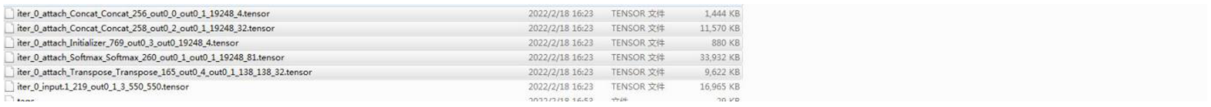
out 目录生成了 yolact 的后处理程序：



```
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$ tree
.
├── Abyssinian_170.jpg
├── dog.jpg
├── Makefile
├── out
│   ├── postprocess
│   └── yolact_post.o
├── result.png
├── tensor
│   ├── output0_4_19248_1.dat
│   ├── output0_4_19248_1.dat.bak
│   ├── output1_81_19248_1.dat
│   ├── output1_81_19248_1.dat.bak
│   ├── output2_32_19248_1.dat
│   ├── output2_32_19248_1.dat.bak
│   ├── output3_4_19248.dat
│   ├── output3_4_19248.dat.bak
│   ├── output4_32_138_138_1.dat
│   └── output4_32_138_138_1.dat.bak
└── yolact_post.cpp

2 directories, 17 files
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$
```

图 2-13：npu_postprocess

yolact 模型有五个输出层,inference 阶段生成了 5 个 output tensor.



iter_0_attach_Concat_Concat_256_out0_0_out0_1_19248_4.tensor	2022/2/18 16:23	TENSOR 文件	1,444 KB
iter_0_attach_Concat_Concat_258_out0_2_out0_1_19248_32.tensor	2022/2/18 16:23	TENSOR 文件	11,570 KB
iter_0_attach_Initializer_769_out0_3_out0_19248_4.tensor	2022/2/18 16:23	TENSOR 文件	880 KB
iter_0_attach_Softmax_Softmax_260_out0_1_out0_1_19248_81.tensor	2022/2/18 16:23	TENSOR 文件	33,932 KB
iter_0_attach_Transpose_Transpose_165_out0_4_out0_1_138_138_32.tensor	2022/2/18 16:23	TENSOR 文件	9,622 KB
iter_0_input.1_219_out0_1_3_550_550.tensor	2022/2/18 16:23	TENSOR 文件	16,965 KB

图 2-14：npu_tensor

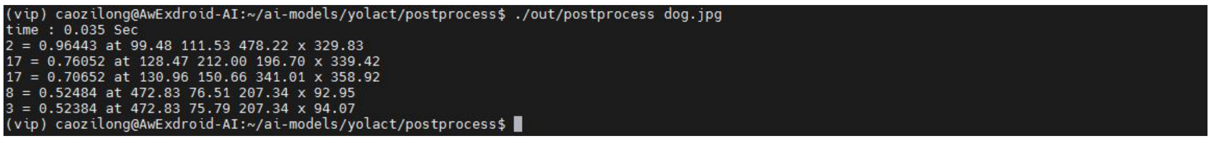
tensor之间的对应关系,0是location，1是confidence，2是mask，3暂时无用，4是maskmaps.

output0_4_19248_1.dat <---->iter_0_attach_Concat_Concat_256_out0_0_out0_1_19248_4.tensor

output1_81_19248_1.dat<---->iter_0_attach_Softmax_Softmax_260_out0_1_out0_1_19248_81.tensor

```
output2_32_19248_1.dat<---->iter_0_attach_Concat_Concat_258_out0_2_out0_1_19248_32.tensor

output3_4_19248.dat  <---->iter_0_attach_Initializer_769_out0_3_out0_19248_4.tensor

output4_32_138_138_1.dat<---->iter_0_attach_Transpose_Transpose_165_out0_4_out0_1_138_138_32.tensor
```

执行后处理



```
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$ ./out/postprocess dog.jpg
time : 0.035 Sec
2 = 0.96443 at 99.48 111.53 478.22 x 329.83
17 = 0.76052 at 128.47 212.00 196.70 x 339.42
17 = 0.70652 at 130.96 150.66 341.01 x 358.92
8 = 0.52484 at 472.83 76.51 207.34 x 92.95
3 = 0.52384 at 472.83 75.79 207.34 x 94.07
(vip) caozilong@AwExdroid-AI:~/ai-models/yolact/postprocess$
```

图 2-15: npu_exepost

后处理结果：

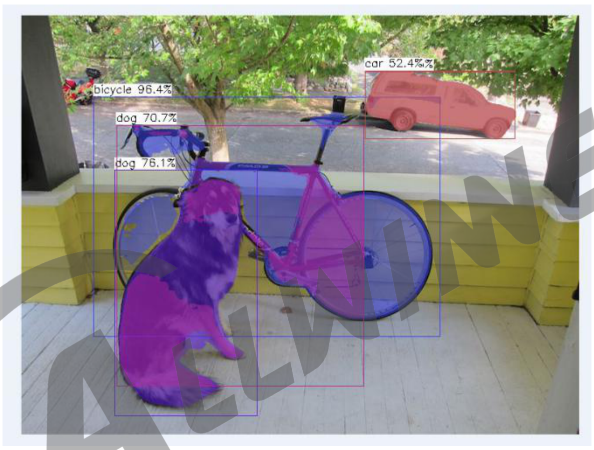


图 2-16: npu_yolact

图中识别除了两只 dog，有两个狗的信息，其他都能很好的输出结果，dog 70.7% 不应该出现的，大概率是量化导致输出结果精度精度降低。

修改归一化参数后重新部署，得到的结果如下，可以看到，之前的问题消失：

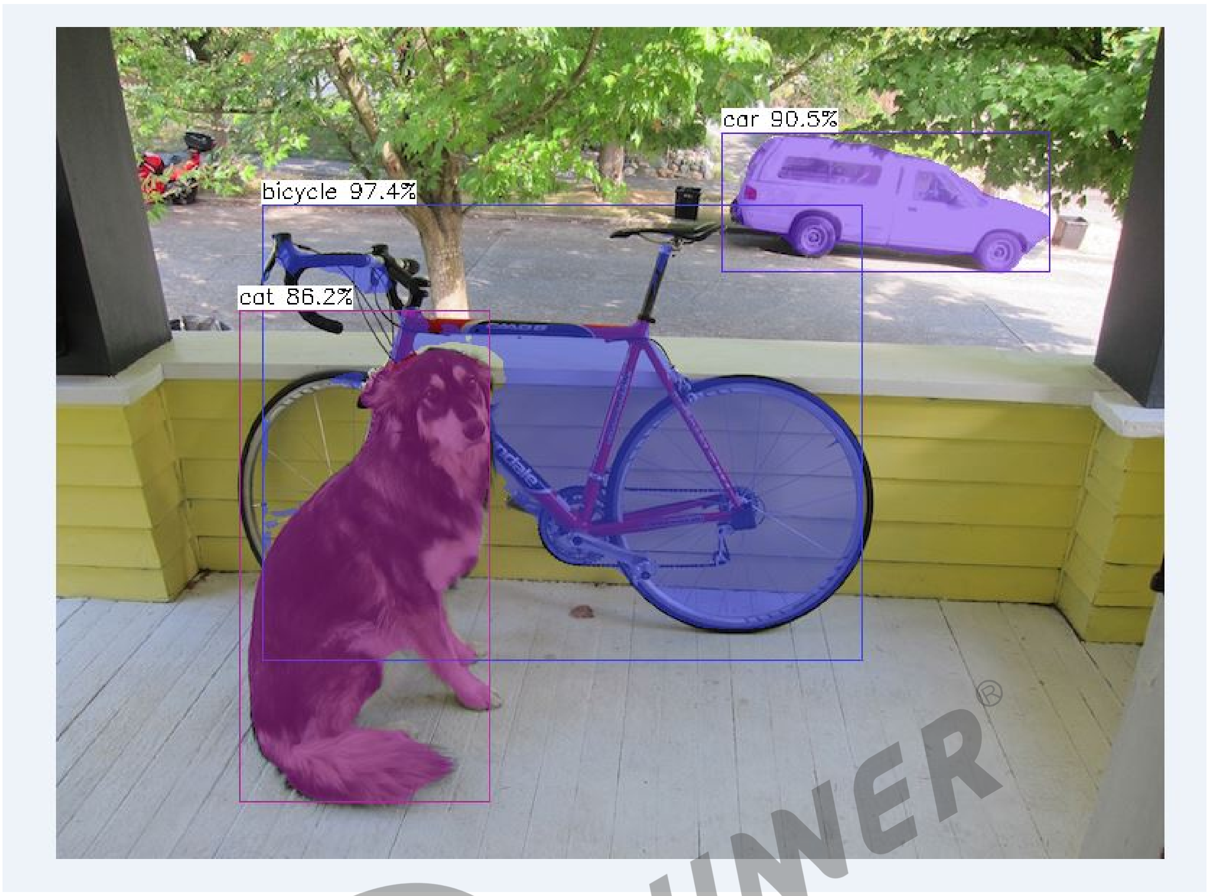


图 2-17: npu_yoloact_ok

至此，yoloact 模型导入完成.

2.12 结束