Ascend 310
V100R001
模型加解密使用指导

文档版本	01
发布日期	2019-03-13

华为技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： http://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

目 录

1 自定义模型加解密

1.1 简介

模型加密功能让模型开发者可以对持有的模型进行加密，从而达到控制模型使用权的目的。模型开发者通过加密功能获取加密后的离线模型和秘钥，同时持有加密后模型和秘钥的用户才能正常使用模型。如果您需要更换密钥或证书，则需要重新对模型加密。

自定义模型加解密的总体流程如图1-1所示。图中灰色底纹部分的操作是由华为工程师执行，其它操作可由用户自行完成。

图 1-1 总体流程

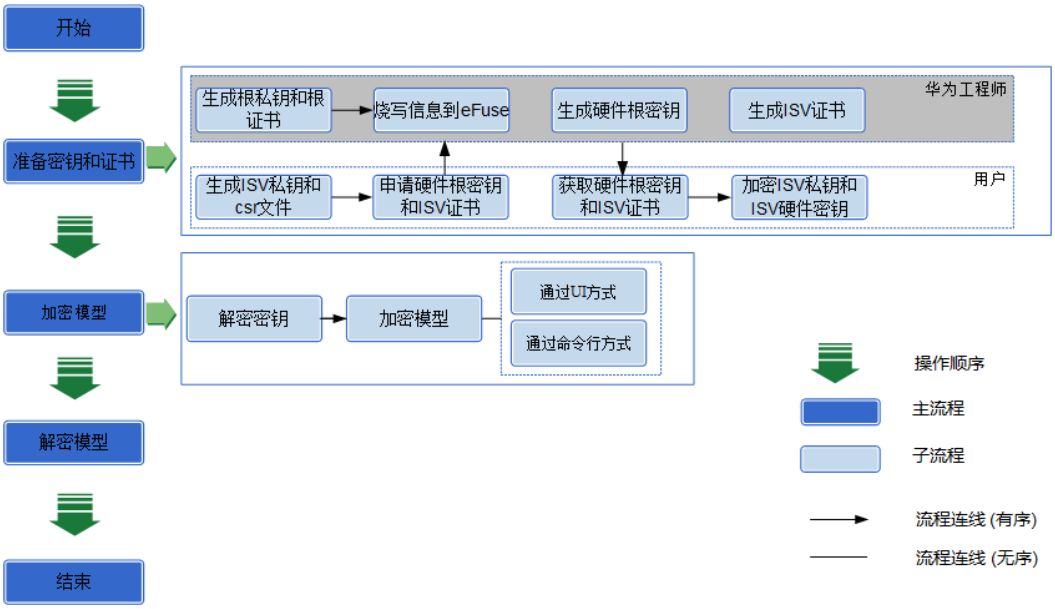


表 1-1 总体流程说明

阶段	步骤	描述
准备密钥和证书	申请ISV硬件密钥和ISV证书	用户使用**openssl**命令生成ISV私钥和csr证书申请文件后，向华为工程师申请ISV硬件密钥和ISV证书。
	加密密钥	为保证密钥安全，建议用户根据实际需求选择加密工具，对ISV硬件密钥、ISV私钥进行加密，存储加密后的密钥。
加密模型	解密密钥	若用户在存储ISV硬件密钥、ISV私钥时，是对密钥加密后再存储的，则在使用密钥前，需要先使用对应的工具解密。
	通过UI或命令行方式加密自定义模型	通过加密自定义模型文件，从而达到控制模型使用权的目的。如果需要替换密钥或证书，则需要通过UI或命令行方式重新对自定义模型进行加密，重新加密时选择新密钥或新证书。 ● 若通过UI方式加密，则需要通过Mind Studio工具新增自定义模型，在参数配置窗口，选择加密模式，并选择ISV硬件密钥、ISV证书、ISV私钥。 ● 若通过命令行方式加密，可以使用**omg**命令加密，在命令行中配置ISV硬件密钥、ISV证书、ISV私钥所在的路径。
解密模型	通过UI方式解密自定义模型	● 通过UI方式加密生成的模型文件，直接在模型组件的属性展示界面选择解密模式，并选择解密文件。 ● 通过命令行方式加密生成的模型文件，暂不支持导入到Mind Studio工具，暂不支持解密。

1.2 申请 ISV 硬件密钥和 ISV 证书

操作步骤

步骤1 执行以下命令，生成ISV私钥。

openssl genrsa -out isv_prikey.key 4096

步骤2 执行以下命令，生成isv_root证书申请文件。

openssl req -new -key isv_prikey.key -sha256 -out isv_req.csr

步骤3 向华为工程师提供isv_req.csr文件，申请ISV硬件密钥、ISV证书。

华为工程师生成ISV硬件密钥、ISV证书提供给用户，用户可使用ISV私钥、ISV硬件密钥、ISV证书执行模型加密。

----结束

1.3 加密自定义模型

1.3.1 通过 UI 方式

1.3.1.1 新增自定义模型组件

自定义模型组件添加有两种方式：

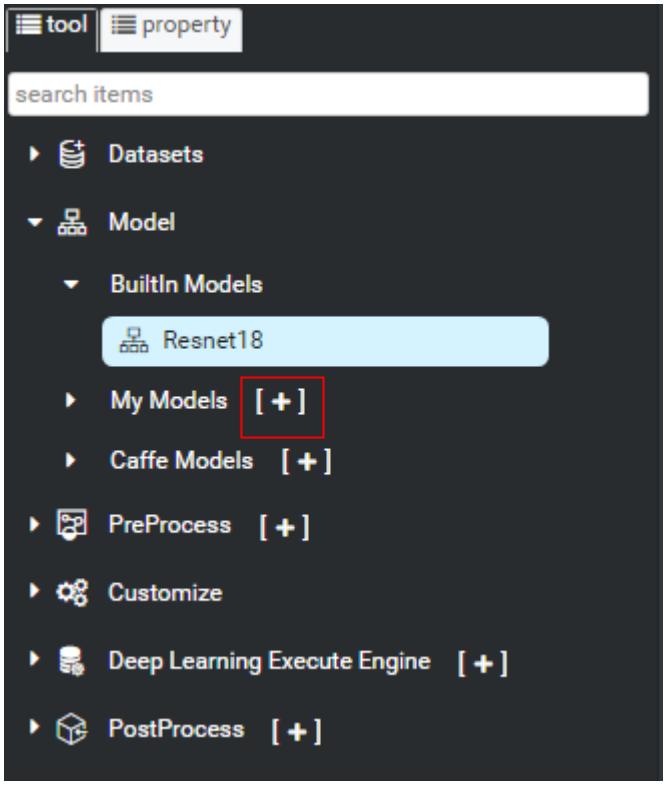
- 在.mind文件的编排窗口中，通过“tool > My Models”新增的方式生成。
- 选中工程后，通过右击选择“Convert Model...”或者界面选择“Tools > Convert Model...”模型转化的方式生成。

下面分别介绍自定义模型组件的两种生成方式。

通过新增方式增加自定义模型组件

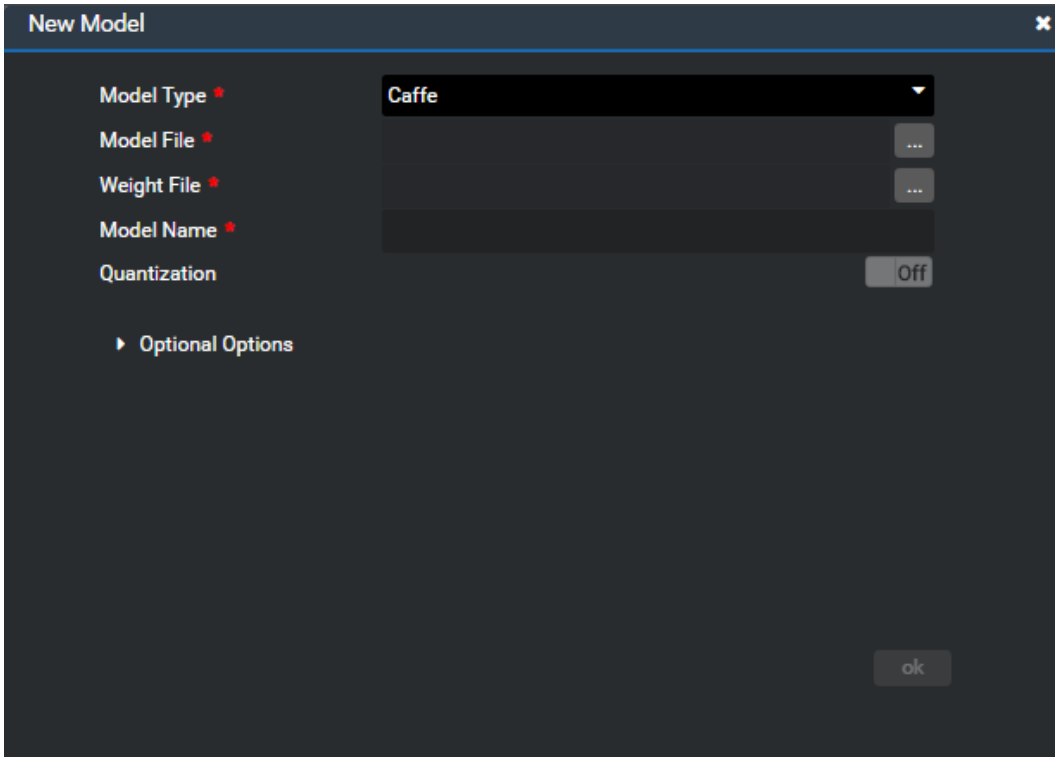
步骤1 单击My Models右侧的[+]，添加自定义模型组件，如图1-2所示。

图 1-2 添加自定义模型



步骤2 弹出New Model编辑窗口，如图1-3所示。

图 1-3 New Model 窗口



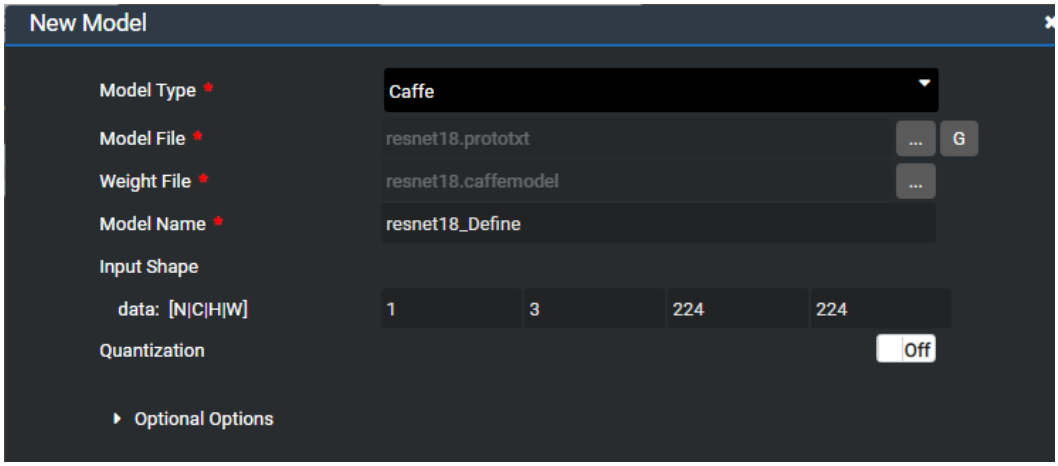
“Model type”可选择Caffe，Tensorflow或OfflineModel：

- Caffe模型：必须配置模型文件 （Model File）和权重文件（Weight File）。
- Tensorflow模型：必须配置模型文件（Model File）。
- OfflineModel：必须选择至少一个模型路径（Device Model Path 或 Emulator Model Path）。

步骤3 单击Model File右侧的  按钮，选择一个模型文件。

如图1-4所示。

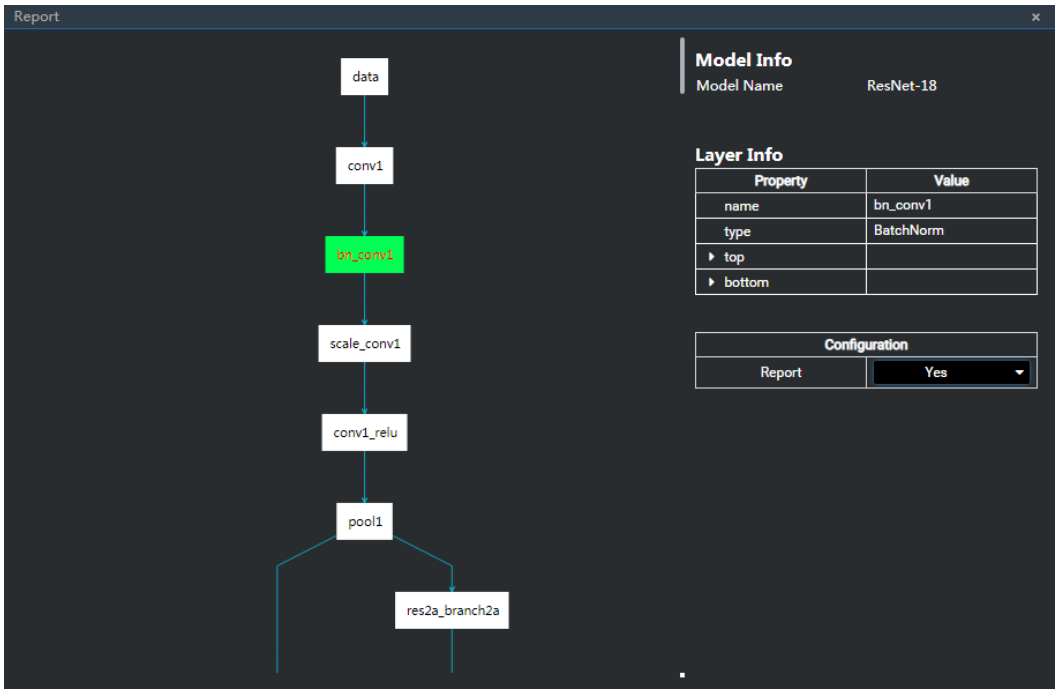
图 1-4 选择模型文件



- “Model Name”自动填充模型文件的名称，用户可以在选择模型文件后自行修改为想要的名称。
- 工具会解析模型文件获取模型的默认“Input Shape”（Tensorflow或者是带有自定义层的模型暂不支持解析，Input Shape以文本输入框呈现，用户自行输入input shape内容，界面不做输入控制，Caffe模型的格式为 input_name:n,c,h,w；Tensorflow模型的格式为 input_name:n,h,w,c；多个input的需要用";"分隔。示例："data1:1,3,224,224;data2:1,3,32,32" ）。

步骤4 选择了模型文件后，Model File右侧会多出G按钮，单击该按钮，展示该模型的原始网络结构图，并且可以在这里设置需要Report（转换后，选中层的输出会直接作为离线模型的输出）的层。如果某层的Report设置为Yes，该层变成绿色。如图1-5所示。

图 1-5 模型网络结构



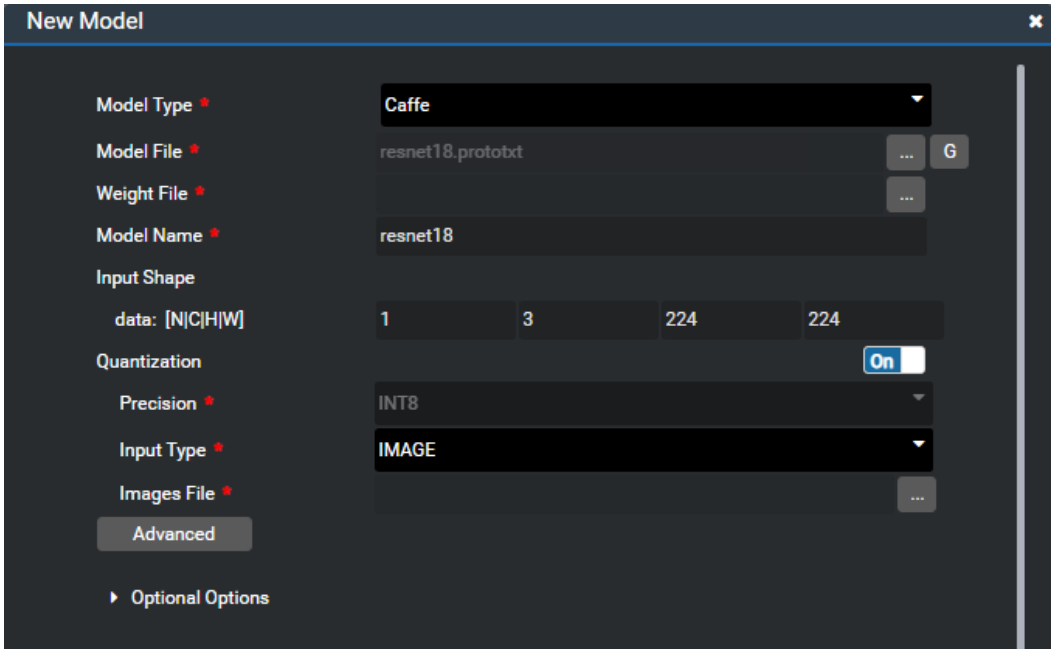
步骤5 打开“Quantization”开关，可以进行量化设置。

“Input Type”可以选择“IMAGE”或者“BINARY”，

- 如果选择IMAGE，Images File选择图片文件夹。
- 如果选择BINARY，Images File选择bin文件。

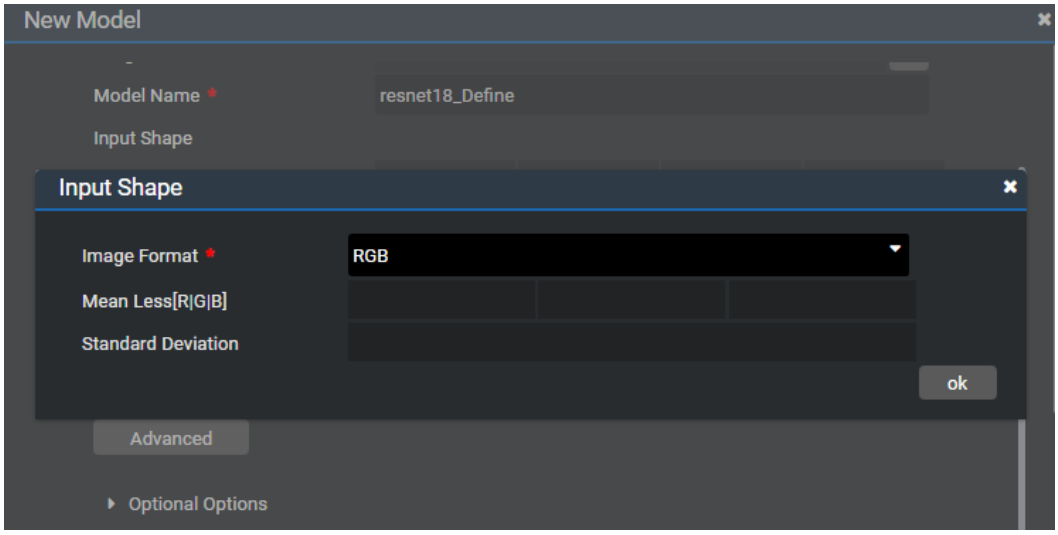
建议选择不超过50张图片进行量化，否则有可能因为量化时间太长导致进程超时（3小时）。如图1-6所示。

图 1-6 量化配置



步骤6 单击“Advanced”，可以设置量化的“Image Format”、“均值”和“标准差”，设置完成后单击“ok”使设置生效，如图1-7所示。

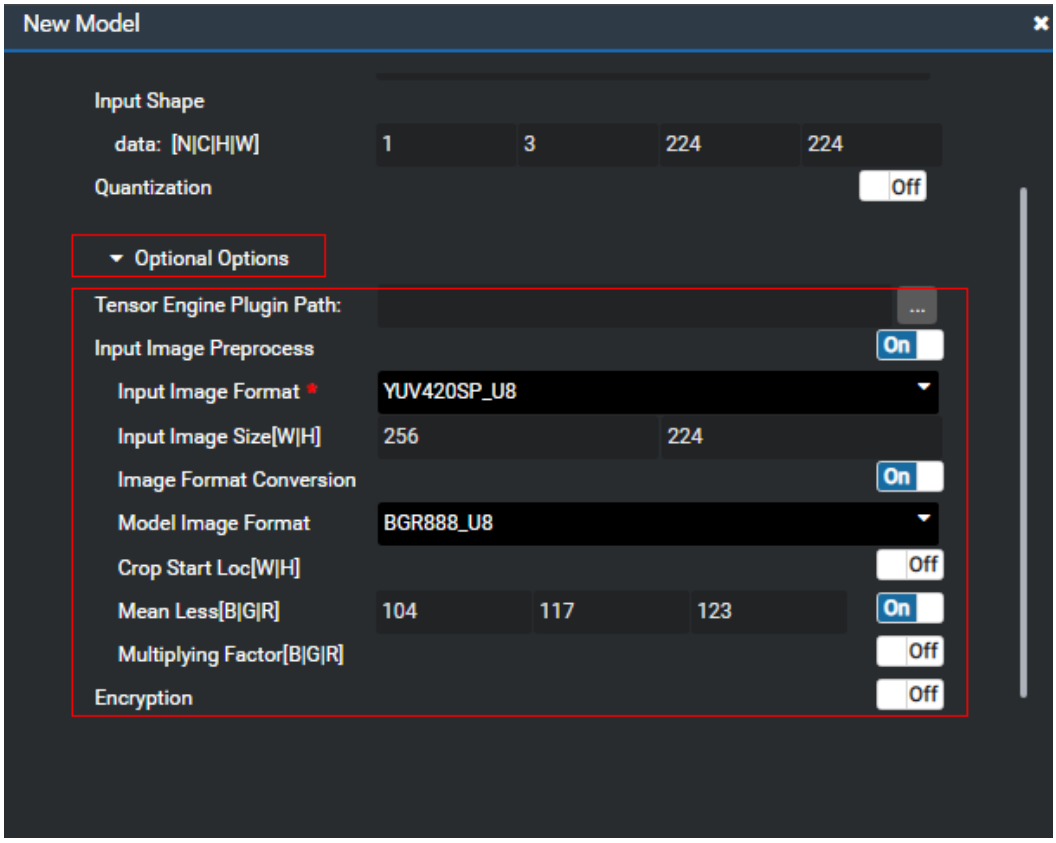
图 1-7 量化高级配置



如果量化开关开启，模型转换完毕后，在相应工程下的convertModel.log文件中可以查看量化参数的配置信息，量化配置的详细说明请参见**1.5.2 量化配置**。

步骤7 单击“Optional Options”，可进行更多可选配置，如图1-8所示。

图 1-8 可选配置



配置项说明如表1-2所示。

表 1-2 Optional Options 参数说明

参数名称	参数描述
Tensor Engine Plugin Path	Tensor Engine算子插件路径。 如果有针对所导入模型的自定义开发算子，此处需要导入。
Input Image Preprocess	Aipp图片预处理相关配置，默认开启，如果不需要设置可以将该开关关闭；如果开启该参数，则模型转换完毕后，在相应工程目录下的**convertModel.log**文件中可以查看相关参数配置信息。
Input Image Format	输入图片格式，默认为YUV420SP_U8。 选项：YUV420SP_U8、XRGB8888_U8、RGB888_U8。
Input Image Size[W|H]	输入图片大小，默认值由模型文件Input层的宽和高分别128和16对齐得到。
Image Format Conversion	色域转换开关，默认开启。 当输入图片格式与模型处理文件格式不一致时需要开启。

参数名称	参数描述
Model Image Format	模型处理图片格式，默认为BGR888_U8。 选项：YUV444SP_U8，YVU444SP_U8，RGB888_U8，BGR888_U8，GRAY 开启色域转换开关后选择。
Crop Start Loc[W|H]	抠图开始位置，默认关闭，开启开关后可以设置起始位置。
Mean Less	减均值，默认开启。 三个通道的值默认为104、117、123。
Multiplying Factor	乘系数（方差或(max-min)的倒数），默认关闭。
Encryption	是否加密。开关打开表示加密，否则为不加密。 具体配置请参见**1.3.1.2 加密自定义模型**。

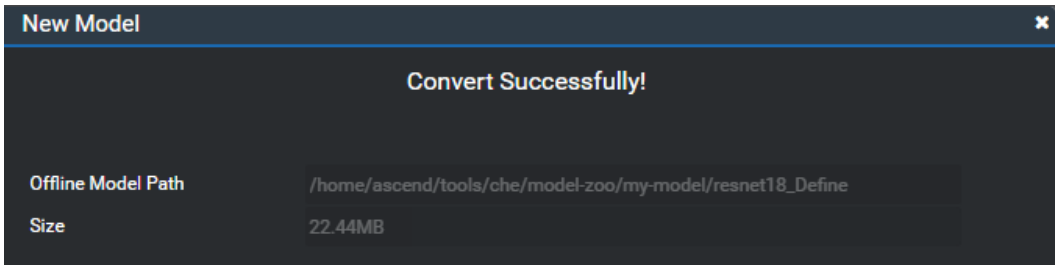
说明

转换模型时开启**Input Image Preprocess**，Input Image Format选择XRGB8888_U8或者RGB888_U8，并且关闭Image Format Conversion，使用这样配置转换出的模型，数据集应当导入格式为NHWC的图片。

步骤8 根据用户需求完成配置后，单击“ok”可以进行模型新建。该操作设置了3小时运行时间限制，如果超过3小时还不能完成模型转换，则会结束转换进程。

新建成功后展示转换成功页面，包括模型路径（路径默认展示在设备运行的模型的路径）和文件大小，如**图1-9**所示。

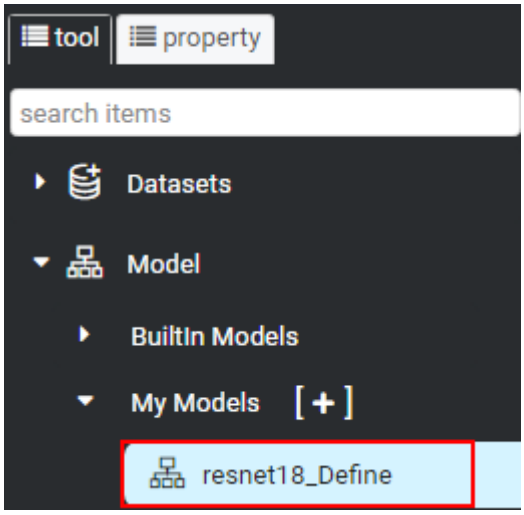
图 1-9 新建模型成功界面



如果失败并且弹出Error Report界面，请参见《Mind Studio快速入门》中的*离线模型转换*章节。

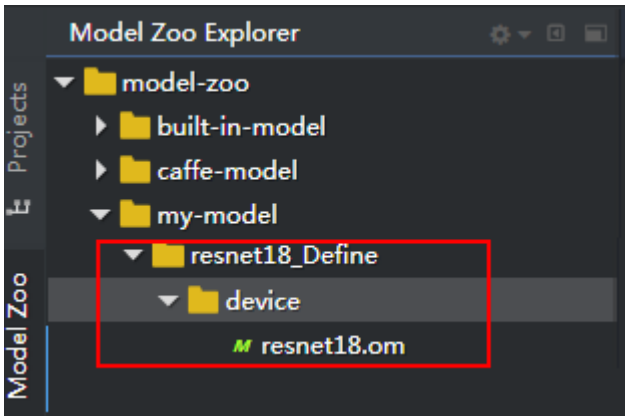
新增模型成功后，会在自定义模型组件中，可直接拖拽用于后续编排，如**图1-10**所示。

图 1-10 自定义模型组件



单击右侧的Model Zoo页签，可以在model-zoo的my-model中看到自定义的模型组件，如图1-11所示。

图 1-11 转化后的模型文件



说明

resnet18.om即为转换生成的模型文件。

----结束

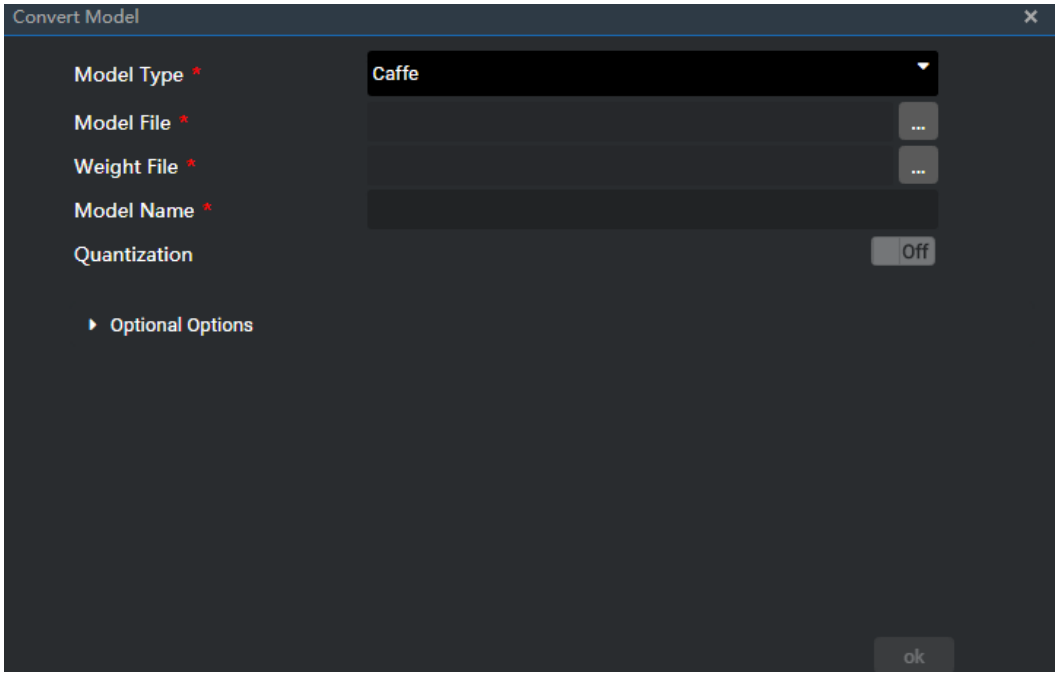
通过模型转换方式生成

步骤1 在Projects Explorer页签中选中需要转化模型的工程名称。

步骤2 右击选择“Convert Model...”或者界面选择“Tools > Convert Model...”。

弹出模型转化配置界面，如图1-12所示。

图 1-12 模型转化配置界面

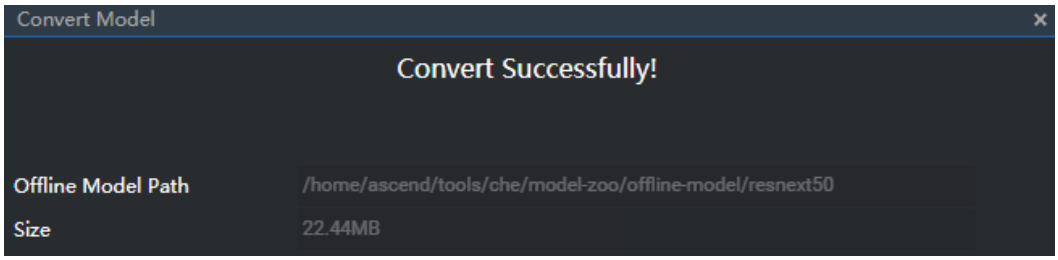


步骤3 模型转化配置界面参数与新建模型参数配置相同，具体请参考**通过新增方式增加自定义模型组件**。

步骤4 配置完成后，单击“ok”，进行模型转化。

步骤5 转换成功后展示转换成功页面，包括模型路径（路径默认展示在设备运行的模型的路径）和文件大小，如**图1-13**所示。

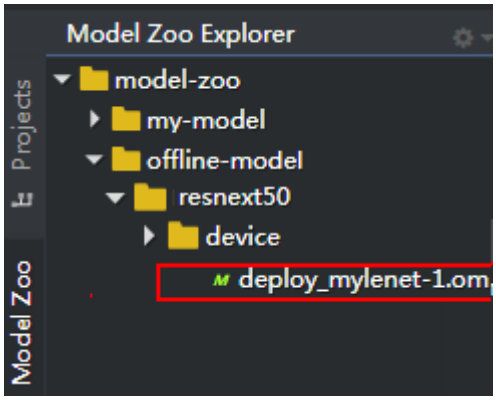
图 1-13 模型转换成功界面



如果失败并且弹出Error Report界面，请参考《Mind Studio快速入门》中的*离线模型转换*章节。

步骤6 模型转化成功后可以在“model-zoo > offline-model”中看到自定义的模型组件，如**图1-14**所示。

图 1-14 转化后的模型文件



步骤7 转化成功后，可以将转化后的模型组件添加到自定义模型组件中供后续拖拽编排使用。

1. 单击“tool > My Models”后的“+”，弹出新建组件窗口。界面参数解释如**表1-3**所示，导入后的界面如**图1-15**所示。

图 1-15 导入自定义模型组件

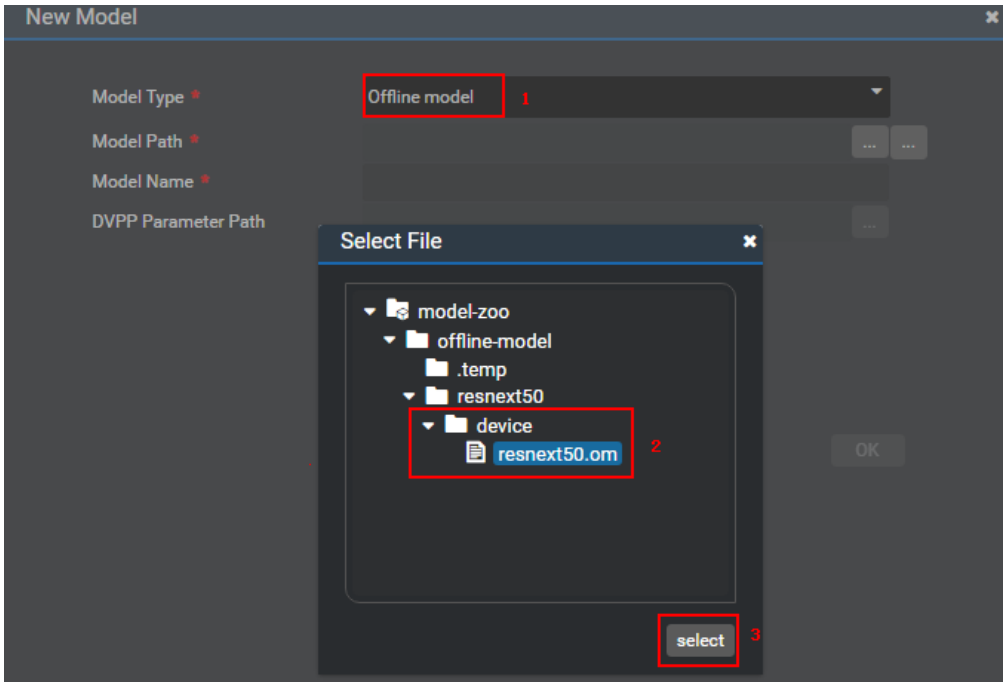


表 1-3 离线模型导入界面参数

界面参数	取值
“Model Type”	选择“OfflineModel”。

界面参数	取值
“Model Path”	选择转化后的自定义单板“device”中的.om模型文件。 左侧  代表从model-zoo的offline-model文件夹中选择模型；右侧  代表从客户端上传模型，此处选择model-zoo。
“Model Name”	根据模型文件名称自动填充。

说明

暂不支持加密模型的导入。

选择完毕后单击“OK”。

2. 单击“select”，将转化后的离线模型组件添加到自定义模型组件栏中。

----结束

1.3.1.2 加密自定义模型

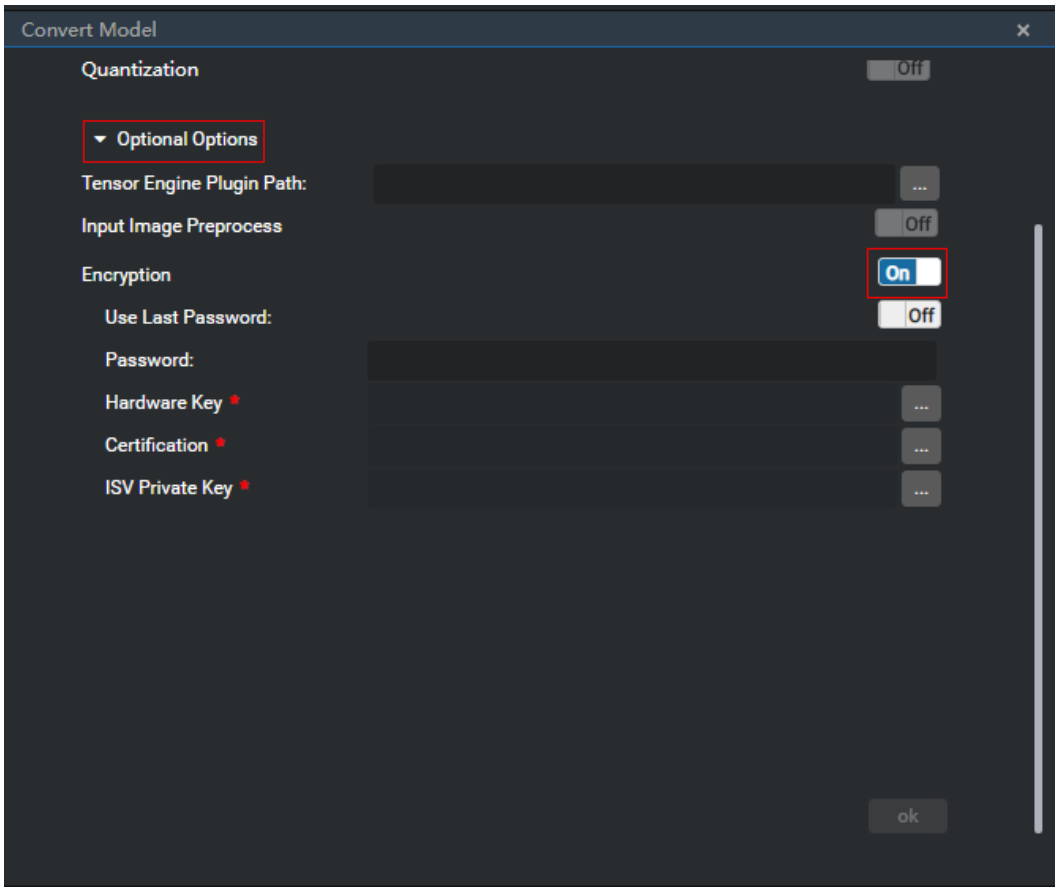
加密的模型仅支持在“Target”为“ASIC”或“Atlas DK”的Mind工程中运行。

模型加密

步骤1 选择工程，在Tools菜单选择“Convert Model”，或者单击开发界面中tool页签中“My Models”的  。

步骤2 单击“Optional Options”，在弹出的界面中激活Encryption选项，如图1-16所示。

图 1-16 激活 Encryption 选项



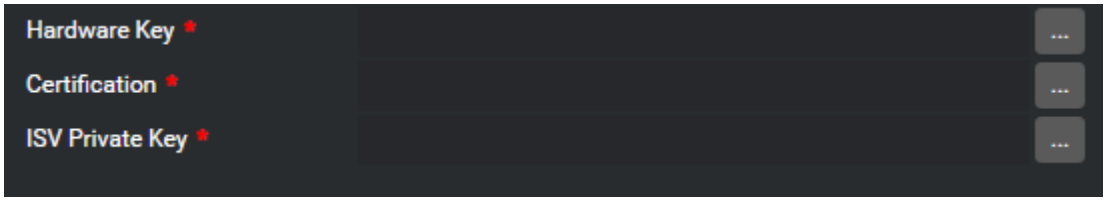
步骤3 输入密码，加密系统会用该密码生成唯一的key，作为加密的输入之一。

步骤4 如果该Workspace曾经输入过密码，则默认用上次的密码，用户可以取消该选项输入新密码，如图1-17所示。

图 1-17 使用上次密码

步骤5 上传ISV硬件密钥、ISV证书以及ISV私钥，如图1-18所示。

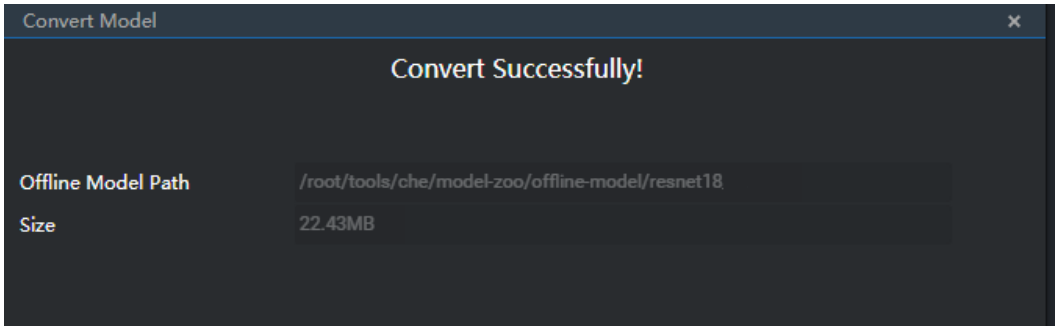
图 1-18 上传密钥、证书以及私钥



步骤6 模型转换完成后，会生成加密后的模型文件、加密后的用于解密模型文件的PASSCODE文件。

模型转换成功后，界面会弹出提示框，提示转换成功，并在“Offline Model Path”处提示加密后的文件的路径，如图1-19所示。

图 1-19 模型转换成功



登录Mind Studio工具所在的服务器，在“Offline Model Path”处的路径下可查看加密后的模型文件、加密后的用于解密模型文件的PASSCODE文件。

----结束

1.3.2 通过命令行方式

1.3.2.1 使用 omg 命令加密

命令功能

转换离线模型。

转换离线模型支持加密、不加密两种场景。不加密场景存在风险，没有使用加密签名，不推荐使用；推荐使用加密方式。若使用非加密方式，framework不再对模型进行完整性校验；这部分完整性由用户在部署应用程序时进行完整性校验，用户保证。

参数说明

参数名称	参数描述	是否必选(以mode为0和3为准)	默认值
--mode	运行模式 ● 0：生成davinci模型 ● 1：模型转json ● 3：仅做预检	否	0
--model	原始框架模型文件路径。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	是	不涉及

参数名称	参数描述	是否必选(以mode为0和3为准)	默认值
--weight	权值文件路径。 当原始模型是caffe时需要指定。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	否	不涉及
--framework	原始框架类型 ● 0：caffe ● 3：tensorflow **说明** mode为1时，支持指定caffe和tensorflow，不指定时为davinci模型转json。	是	不涉及
--output	存放转换后的离线模型文件的路径（包含文件名），例如“out/caffe_resnet18”。 转换后的模型文件，会自动以“.om”的后缀结尾。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	是	不涉及
--encrypt_mode	加密模式 ● 0：加密 ● -1：不加密	否	-1
--encrypt_key	用于加密的随机数文件所在的路径。 加密模式下必填。 **说明** 1、路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)。 2、测试时，您可以使用**openssl rand 32 -out ek_key**命令生成一个随机数文件。实际商用时，用户可根据实际需求选择其它工具生成随机数文件。	否	不涉及

参数名称	参数描述	是否必选(以mode为0和3为准)	默认值
--hardware_key	加密使用的ISV硬件密钥文件路径。 加密模式下必填。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	否	不涉及
--certificate	加密使用的ISV证书文件路径。 加密模式下必填。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	否	不涉及
--private_key	加密使用的ISV私钥文件路径。 加密模式下必填。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	否	不涉及
--aipp_conf	aipp配置文件路径。 **说明** 1、路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.) ● 2、aipp配置文件的内容示例如下： ● input_format: YUV420SP_U8 ● csc_switch：true ● var_reci_chn_0：0.00392157 ● var_reci_chn_1：0.00392157 ● var_reci_chn_2：0.00392157 ● 3、aipp配置文件的配置说明，请参见**1.5.1 AIPP配置说明**。	否	不涉及

参数名称	参数描述	是否必选(以mode为0和3为准)	默认值
--cal_conf	量化配置文件路径。 **说明** 1、路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.) ● 2、量化配置文件的内容示例如下： ● device：USE_CPU ● bin：150 ● type：JSD ● quantize_algo：NON_OFFSET ● inference_with_data_quantized：true ● inference_with_weight_quantized：true ● 3、量化配置文件的配置说明，请参见**1.5.2 量化配置**。	否	不涉及
--check_report	预检结果保存文件路径。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	否	check_result.json
--compress	是否启用压缩。	否	false
--h或--help	显示帮助信息。	否	不涉及
--input_format	输入数据格式：NCHW和NHWC 当原始框架是tensorflow时，默认是NHWC。如果实际是NCHW的话，需要通过此参数指定NCHW。	否	不涉及
--input_fp16_nodes	指定数据类型为“fp16 nc1hwc0”的输入节点名称。 例如：“node_name1; node_name2”。	否	不涉及
--input_shape	输入数据的shape。 例如：“input_name1: n1, c1, h1, w1; input_name2: n2, c2, h2, w2”。	否	不涉及
--is_output_fp16	标注输出的数据类型是否为“fp16 nc1hwc0”。 例如：false, true, false, true。	否	false

参数名称	参数描述	是否必选(以mode为0和3为准)	默认值
--json	模型文件转换为json格式文件的路径。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	否	不涉及
--om	模型文件路径。 当mode为1时必填。 **说明** 路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.)	否	不涉及
--op_name_map	算子映射配置文件路径。 包含DetectionOutput网络时需要指定。 例如：不同的网络中DetectionOutput算子的功能不同，可能指定DetectionOutput到FSRDetectionOutput或者SSDDetectionOutput的映射。 **说明** 1、路径部分：支持大小写字母、数字，下划线；文件名部分：支持大小写字母、数字，下划线和点(.) 2、算子映射配置文件的内容示例如下: DetectionOutput: SSDDetectionOutput。	否	不涉及
--out_nodes	指定输出节点。 例如：“node_name1:0;node_name1:1;node_name2:0”。	否	不涉及
--plugin_path	自定义算子插件路径。 例如："/home/a1/b1;/home/a2/b2;/home/a3/b3" **说明** 自定义算子插件路径中可以包含多个路径，两个路径之间以分号分割，每个路径内不能包含分号，否则将导致解析得到的路径与预期不符。	否	./plugin

参数名称	参数描述	是否必选(以mode为0和3为准)	默认值
--target	目标平台：mini mini：量化中eltwise算子支持双输出；量化中roipooling算子支持int8输出；量化中conv算子支持混合精度。	否	不涉及
--ddk_version	指定TVM自定义算子运行需要匹配的ddk环境的版本号。	否	不涉及
--net_format	指定网络算子优先选用的数据格式，ND和5D。	否	不涉及

使用说明

执行omg命令有以下两种方式：

- 以**HwHiAiUser**用户登录Host侧服务器，切换到root用户后，执行omg命令，本节以此种方式为例。
- 以Mind Studio安装用户登录Mind Studio服务器，切换到“~/tools/che/ddk/ddk/uihost/bin”目录下，先配置环境变量，再执行omg命令。

  配置环境变量，其中，“/home/ascend/tools”需替换为实际的toolpath路径。

  ```
  export LD_LIBRARY_PATH=/home/ascend/tools/che/ddk/ddk/uihost/lib/
  ```

  omg命令执行示例：

  ```
  ./omg --h
  ```

  说明

  “~/tools”是默认的toolpath路径，该路径可在安装Mind Studio时由用户自定义，您可以在“scripts/env.conf”文件通过toolpath参数查看实际路径。您可以使用**find / -name 'env.conf'**命令查看script目录下的“env.conf”文件的位置。

使用示例

步骤1 以**HwHiAiUser**用户登录Host侧服务器。

步骤2 执行如下命令切换到root用户。

su - root

步骤3 执行以下命令生成加密的模型文件。

```
omg --model=/usr/local/HiAI/driver/tools/test/resnet18.prototxt --weight=/usr/local/HiAI/driver/
tools/test/resnet18.caffemodel --framework=0 --output=/usr/local/HiAI/driver/tools/test/out/
caffe_resnet18 --encrypt_mode=0 --encrypt_key=/usr/local/HiAI/driver/tools/test/ek_key --
hardware_key=/usr/local/HiAI/driver/tools/test/isv_hw_key --certificate=/usr/local/HiAI/driver/
tools/test/isv_cert.crt --private_key=/usr/local/HiAI/driver/tools/test/isv_prikey.key
```

成功执行命令后，在output参数指定的路径下，可查看加密后的模型文件（如：caffe_resnet18.om）以及加密后的用于解密模型文件的文件（caffe_resnet18.PASSCODE）。

----结束

1.4 解密自定义模型

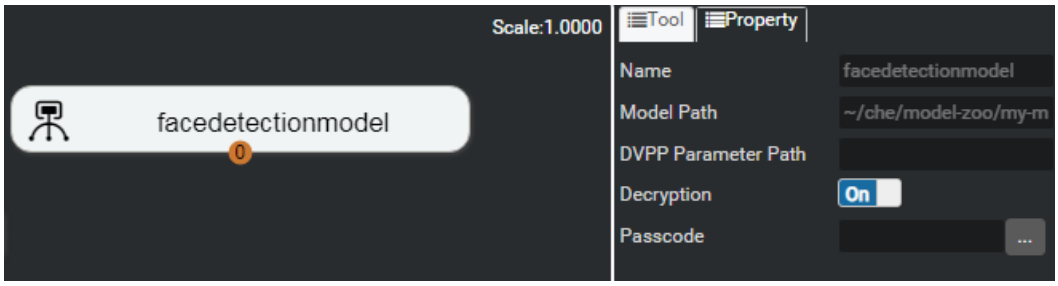
1.4.1 解密模型文件

模型解密

步骤1 把加密后的离线模型添加到可拖拉组件中。

步骤2 把离线模型拖到编排界面中，单击激活模型属性展示界面，如图1-20所示。

图 1-20 模型属性展示页面



步骤3 打开Decryption开关，选择Passcode文件，解密后，可进行正常开发。

----结束

1.5 参考

1.5.1 AIPP 配置说明

如果离线模型转换的时候开启“Input Image Preprocess”开关参数（请参见《Ascend 310 Mind Studio基本操作》中的“模型管理>Mind Engine的模型管理>新增自定义模型组件”中的“Optional Options参数说明”），则在模型转换完毕后，相应工程下的convertModel.log文件中可以查看相应参数的配置信息。

配置文件说明如下：

```
#
# AIPP当前支持色域转换、抠图、减均值、乘系数、通道数据交换、单行模式的能力。
# 输入图片的类型仅支持UINT8格式。
# 使用此配置文件时，请将需要配置的参数去注释，并改为合适的值。
# 模板中参数值为默认值，其中input_format属性为必选属性，其余属性均为可选配置。
#

#======================== 全局设置 ========================

# 输入图像类型
# 类型：enum
# 取值范围：YUV420SP_U8/XRGB8888_U8/RGB888_U8/YUV400_U8
# input_format :

# 色域转换开关
# 类型：bool
# 取值范围：true/false
# csc_switch :false
```

```
# 图像的宽度、高度
# 类型：uint13
# 取值范围 & 约束： (0,4096]、对于YUV420SP_U8类型的图像，要求取值是偶数
# 说明：请根据实际图片的宽、高配置src_image_size_w、src_image_size_h，若不设置或设置为0，则会取网络输入定义的w和h
# src_image_size_w :0
# src_image_size_h :0

# 抠图起始位置水平、垂直方向坐标，抠图大小为网络输入定义的w和h
# 类型：uint13
# 取值范围 & 约束： [0,4096]、对于YUV420SP_U8类型的图像，要求取值是偶数
#说明：load_start_pos_w加上网络输入定义的w需要小于等于src_image_size_w，load_start_pos_h加上网络输入定义的h需要小于等于src_image_size_h
# load_start_pos_w :0
# load_start_pos_h :0

# C方向的填充值
# 类型： float16
# 取值范围：[-65504, 65504]
# cpadding_value :0.0

# 色域转换前，R通道与B通道交换开关/U通道与V通道交换开关
# 类型：bool
# 取值范围：true/false
# rbuv_swap_switch :false

# 色域转换前，RGBA->ARGB, YUVA->AYUV交换开关
# 类型：bool
# 取值范围：true/false
# ax_swap_switch :false

# 单行处理模式（只处理抠图后的第一行）开关
# 类型：bool
# 取值范围：true/false
# single_line_mode :false

# AIPP处理图片时是缩放还是裁剪 （保留字段）
# 类型：bool
# 取值范围：true/false，true表示缩放，false表示裁剪
# resize :true

#======================== 减均值、乘系数设置 ========================
# 计算规则如下：
# 当uint8->uint8时，本功能旁路
# 当uint8->int8时，pixel_out_chx(i) = pixel_in_chx(i) - mean_chn_i
# 当uint8->fp16时，pixel_out_chx(i) = [pixel_in_chx(i) - mean_chn_i - min_chn_i] * var_reci_chn

# 通道n均值
# 类型：uint8
# 取值范围：[0, 255]
# mean_chn_0 :0
# mean_chn_1 :0
# mean_chn_2 :0

# 通道n最小值
# 类型：float16
# 取值范围：[-65504, 65504]
# min_chn_0 :0.0
# min_chn_1 :0.0
# min_chn_2 :0.0

# 通道n方差或(max-min)的倒数
# 类型：float16
# 取值范围：[-65504, 65504]
# var_reci_chn_0 :1.0
# var_reci_chn_1 :1.0
# var_reci_chn_2 :1.0
```



```
#======================== 色域转换参数设置 ========================
# 若色域转换开关为false，则本功能旁路。
# 若输入图片通道数为4，则忽略第一通道。
# YUV转BGR：
# | B |   | matrix_r0c0 matrix_r0c1 matrix_r0c2 | | Y - input_bias_0 |
# | G | = | matrix_r1c0 matrix_r1c1 matrix_r1c2 | | U - input_bias_1 | >> 8
# | R |   | matrix_r2c0 matrix_r2c1 matrix_r2c2 | | V - input_bias_2 |
# BGR转YUV：
# | Y |   | matrix_r0c0 matrix_r0c1 matrix_r0c2 | | B |         | output_bias_0 |
# | U | = | matrix_r1c0 matrix_r1c1 matrix_r1c2 | | G | >> 8 + | output_bias_1 |
# | V |   | matrix_r2c0 matrix_r2c1 matrix_r2c2 | | R |         | output_bias_2 |

# 3*3 CSC矩阵元素
# 类型：int16
# 取值范围：[-32768 ,32767]
# matrix_r0c0 :298
# matrix_r0c1 :516
# matrix_r0c2 :0
# matrix_r1c0 :298
# matrix_r1c1 :-100
# matrix_r1c2 :-208
# matrix_r2c0 :298
# matrix_r2c1 :0
# matrix_r2c2 :409

# RGB转YUV时的输出偏移
# 类型：uint8
# 取值范围：[0, 255]
# output_bias_0 :16
# output_bias_1 :128
# output_bias_2 :128

# YUV转RGB时的输入偏移
# 类型：uint8
# 取值范围：[0, 255]
# input_bias_0 :16
# input_bias_1 :128
# input_bias_2 :128
```

1.5.2 量化配置

如果离线模型转换的时候开启“Quantization”开关参数（请参见《Ascend 310 Mind Studio基本操作》中的“模型管理 > Mind Engine的模型管理 > 新增自定义模型组件”中的参数解释），则在模型转换完毕后，相应工程下的convertModel.log文件中可以查看相应参数的配置信息。

配置文件格式

- 现在支持Convolution、Full Connection算子不带offset、半带offset场景下的权重、偏置和数据量化，其中权重支持SCALAR和VECTOR模式。
- 支持网络多输入的场景，输入支持图片（IMAGE）和二进制文件（BINARY）两种格式：
  - 图片（IMAGE格式）支持jpg、png、bmp格式。
  - 二进制文件（BINARY格式）格式详情见表1-4。

表 1-4 二进制文件格式

Offset	type	value	description
0000	32bit int	magic	magic number
0004	32bit int	50	input num

Offset	type	value	description
0008	32bit int	3	input channels
0012	32bit int	28	input height
0016	32bit int	28	input width
…	float	126	pixel

说明

BINARY的头有20bytes，5个int数值，第一个magic_num = 510，用来做校验。另外4个数是n、c、h、w。后面是所有数据，数据是float类型，数据数量等于n*c*h*w。对于非四维的数据，需要补齐到四维，补齐的维度为1。

配置文件模板

量化配置文件根据用户需要命名，其中内容参考以下模板（使用英文格式）。

- 单输入场景（输入为图片）。

```
device:USE_CPU
quantize_algo:HALF_OFFSET
weight_type:VECTOR_TYPE
preprocess_parameter:
{
input_type:IMAGE
image_format:BGR
input_file_path:'calibration/image_set'
mean_value:104.0
mean_value:117.0
mean_value:123.0
standard_deviation:1.0
}
```

- 多输入场景（第一个输入为图片，第二个输入为二进制文件）。

```
device:USE_CPU
quantize_algo:HALF_OFFSET
weight_type:VECTOR_TYPE
preprocess_parameter:
{
input_type:IMAGE
image_format:BGR
input_file_path:'calibration/image_set'
mean_value:104.0
mean_value:117.0
mean_value:123.0
standard_deviation:1.0
}
preprocess_parameter:
{
input_type:BINARY
input_file_path:'calibration/img_info.bin'
}
```

- 配置参数大全版（多输入场景：第一个输入为图片，第二个输入为二进制文件）。

```
device:USE_CPU
quantize_algo:HALF_OFFSET
weight_type:VECTOR_TYPE
bin:150
type:KL
inference_with_data_quantized:false
inference_with_weight_quantized:true
```



```
super_parameter:
{
min_percentile:PERCENTILE_HIGH
max_percentile:PERCENTILE_MID
start_ratio:0.7
end_ratio:1.3
step_ratio:0.01
}
exclude_op:'fc1000'
batch_count:50
preprocess_parameter:
{
input_type:IMAGE
image_format:BGR
input_file_path:'calibration/image_set'
mean_value:104.0
mean_value:117.0
mean_value:123.0
standard_deviation:1.0
}
preprocess_parameter:
{
input_type:BINARY
input_file_path:'calibration/img_info.bin'
}
```

说明

使用此配置文件时，请将需要的参数改成合适的值。以上模板中参数只是建议值，通常情况建议值不需要修改（输入文件除外）。

配置文件参数说明

全局设置

表 1-5 device 参数说明

名称	推理模式
类型	enum
取值范围	USE_CPU
参数意义	USE_CPU：使用CPU做推理。
说明	当前量化使用CPU做推理。
推荐配置	USE_CPU

表 1-6 quantize_algo 参数说明

名称	量化模式
类型	enum
取值范围	NON_OFFSET/HALF_OFFSET

参数意义	量化的映射公式有两种： ● 一种带偏移，公式为：q_uint8 = round(d_float/scale) - offset ● 一种不带偏移，公式为：q_int8 = round(d_float/scale) 涉及映射规则的量化数据有两块：权重、数据。 ● NON_OFFSET表示：权重和数据都采用不带偏移模式 ● HALF_OFFSET表示：数据采用带偏移模式，权重采用不带偏移模式
说明	当前支持NON_OFFSET和HALF_OFFSET模式。
推荐配置	HALF_OFFSET

表 1-7 weight_type 参数说明

名称	权重量化模式
类型	enum
取值范围	VECTOR_TYPE/SCALAR_TYPE
参数意义	对于卷积算子，可能有多个卷积核，多个对应的量化参数可能不一样。 ● 当配置成VECTOR_TYPE时，表示一个卷积核对应一组量化参数 ● 当配置成SCALAR_TYPE时，表示多个卷积核使用同一组量化参数
说明	推荐使用VECTOR_TYPE模式
推荐配置	VECTOR_TYPE

表 1-8 preprocess_parameter 参数说明

名称	预处理及输入相关参数
类型	Struct
取值范围	内部包含input_type、image_format、input_file_path、mean_value以及standard_deviation
参数意义	指定预处理及输入相关参数。
说明	preprocess_parameter参数的数量与网络的输入算子数量必须相同。
推荐配置	无。

表 1-9 input_type 参数说明

名称	输入数据类型

类型	enum
取值范围	IMAGE/BINARY
参数意义	指定输入类型： ● IMAGE：图片格式 ● BINARY：二进制格式
说明	根据需要配置。
推荐配置	无。

表 1-10 image_format 参数说明

名称	图像输入数据三通道排序方式
类型	enum
取值范围	BGR/RGB
参数意义	模型训练时候的图片三通道排序格式
说明	输入为图片的时候需要配置该值。该参数根据实际网络训练时候的输入通道排序方式确定，通常网络训练时候的排序方式是BGR。
推荐配置	BGR

表 1-11 input_file_path 参数说明

名称	输入数据地址
类型	string
取值范围	路径不要有中文、特殊字符以及空格。
参数意义	指定量化输入的目录或文件。
说明	根据实际网络的输入配置。 ● 如果是图片，指定到目录 ● 如果是二进制文件，指定到文件
推荐配置	建议使用与应用场景相关的图片或二进制文件。

表 1-12 mean_value 参数说明

名称	图像预处理的均值
类型	float
取值范围	[0, 255.0]

参数意义	图片预处理单个通道的均值。
说明	输入为图片的时候需要配置该值。多个通道需要配置多个该参数，通常图片有RGB三个通道，则需要配置三个该参数，如下所示。 mean_value：104.0 mean_value：117.0 mean_value：123.0
推荐配置	无。

表 1-13 standard_deviation 参数说明

名称	图像预处理的方差
类型	float
取值范围	[0,FLOAT_MAX]
参数意义	图片预处理的方差。
说明	输入为图片的时候需要配置该值。多个通道使用统一方差。 如果输入范围大于float型能表示的范围，模型量化精度不能保证。 如果0<=standard_deviation<=0.00001，standard_deviation取1.0。
推荐配置	1.0（图片取值区间大小不发生变化）或者255.0（对于图片区间大小为[0, 255]的场景，可将值压缩到[0, 1.0]）。

表 1-14 bin 参数说明

名称	数据映射直方图范围
类型	uint32
取值范围	[0,1000]
参数意义	数据直方图统计的范围。
说明	在计算散度过程中需要统计数据直方图，该参数的值决定了直方图统计的最大值。如果不配置或者配置为0，则使用默认配置150。
推荐配置	100/150/200/250。

表 1-15 type 参数说明

名称	散度计算指标类型
类型	enum
取值范围	KL/SYMKL/JSD

参数意义	KL：Kullback-Leibler Divergence；SYMKL：Symmetric Kullback-Leibler Divergence；JSD：Jensen-Shannon Divergence。
说明	不同的散度类型对应的计算方式不一样。默认为KL。
推荐配置	无。

表 1-16 inference_with_data_quantized **参数说明**

名称	推理过程中是否使用量化后的输入数据
类型	bool
取值范围	true/false
参数意义	控制推理过程中的输入数据是否经过量化。
说明	该参数开启，模拟了输入数据量化的过程。默认值为false。
推荐配置	false

表 1-17 inference_with_weight_quantized **参数说明**

名称	推理过程中是否使用量化后的权重数据
类型	bool
取值范围	true/false
参数意义	控制推理过程中的权重数据是否经过量化。
说明	该参数开启，模拟了权重数据量化反量化的过程。默认值为true。
推荐配置	true

表 1-18 super_parameter **参数说明**

名称	搜索相关参数
类型	Struct
取值范围	内部包含min_percentile、max_percentile、start_ratio、end_ratio以及step_ratio
参数意义	搜索相关参数。
说明	建议使用默认配置。
推荐配置	无。

表 1-19 min_percentile 参数说明

名称	最小值搜索位置
类型	enum
取值范围	PERCENTILE_HIGH/PERCENTILE_MID/PERCENTILE_LOW
参数意义	决定取第多少小的数，比如有100个数，1.0表示取第100-100*1.0=0，对应的就是第一个小的数。
说明	PERCENTILE_HIGH：1.0 PERCENTILE_MID：0.99999 PERCENTILE_LOW：0.9999
推荐配置	PERCENTILE_HIGH

表 1-20 max_percentile 参数说明

名称	最大值搜索位置
类型	enum
取值范围	PERCENTILE_HIGH/PERCENTILE_MID/PERCENTILE_LOW
参数意义	决定取第多少大的数，比如有100个数，1.0表示取第100-100*1.0=0，对应的就是第一个大的数。
说明	PERCENTILE_HIGH：1.0 PERCENTILE_MID：0.99999 PERCENTILE_LOW：0.9999
推荐配置	PERCENTILE_MID

表 1-21 start_ratio、end_ratio、step_ratio 参数说明

名称	参数说明
类型	float/float/float
取值范围	end_ratio>start_ratio>0 && step_ratio>0（如果输入范围大于float型能表示的范围，模型量化精度不能保证）
参数意义	● start_ratio：决定搜索开始的位置 ● end_ratio：决定搜索结束的位置 ● step_ratio：决定搜索步长

说明	在算法中找到d_max/d_min之后，会根据此参数，取d_max/d_min前后多少范围之内的数，然后根据step_ratio决定，每次增加的步长，说明参考如下： 以d_max =100，start_ratio=0.8，end_ratio=1.2，step_ratio=0.01为例，其定义的d_max搜索空间为从100*0.8=80到100*1.2=120的范围，每次步进100*0.01=1，一共41个d_max搜索值。
推荐配置	推荐配置有两组 ● start_ratio:0.7 end_ratio:1.3 step_ratio:0.01 ● start_ratio:0.3 end_ratio:1.7 step_ratio:0.01

表 1-22 batch_count 参数说明

名称	量化校准集图片处理的批次数量
类型	uint32
取值范围	[0,UINT32_MAX)
参数意义	决定量化的读取的校准集图片数量。
说明	如果不配置或者配置为0，那么会把校准集所有图片都作为校准集数据。如果配置了大于0的数，那么会根据校准集路径下图片总数和该参数取较小值作为校准集的图片实际数量。建议校准集图片数量不超过500张。
推荐配置	无。

表 1-23 exclude_op 参数说明

名称	量化算子黑名单
类型	string
取值范围	算子名称
参数意义	配置该参数，算子不进行量化
说明	1. 只支持Conv、FC、DepthWiseConv算子； 2. 多个算子，需要配置多个该参数，每个参数一行，如： exclude_op:'aaa' exclude_op:'bbb'
推荐配置	无