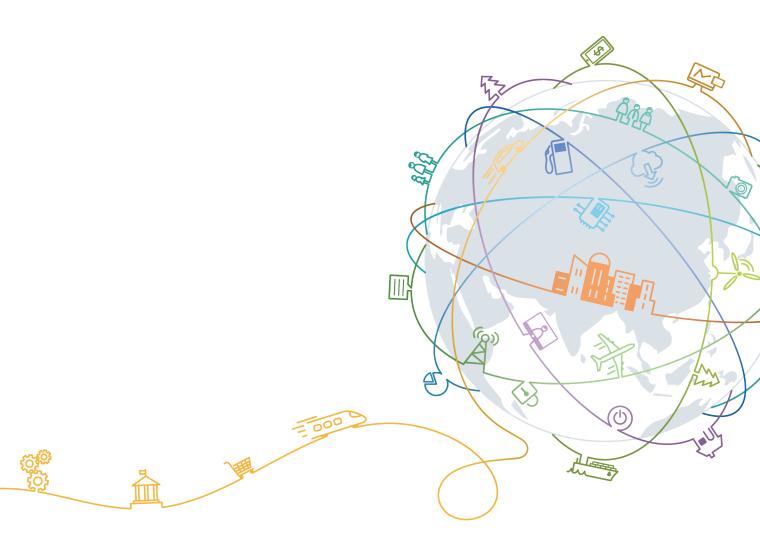
Ascend 310 V100R001

Mind Studio 基本操作

文档版本 01

发布日期 2019-03-13





版权所有 © 华为技术有限公司 2019。 保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部,并不得以任何形式传播。

商标声明



HUAWEI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 概述	1
1.1 简介	1
1.2 功能说明	2
2 用户管理	Δ
2.1 用户登录	
2.2 重置密码	
2.3 修改密码	
2.4 用户退出	
2.5 安全日志查询	
3 工程管理	
3 1 简介	
3.2 基本操作	
3.2.1 工程的建/删除	
3.2.3 新建/删除文件	
3.2.4 文件/文件夹上传	
3.2.5 打开项目	
3.2.6 关闭项目	
3.2.7 工程编译	
3.2.8 工程运行	
3.2.9 支持的文件格式	
4 数据集管理	21
- 30 A A B - 2	
4.2 Mind Engine 的数据集管理	
4.2.1 数据集展示	
4.2.2 数据集导入	
4.2.3 数据集属性展示	
4.2.4 生成 cpp 文件	40
4.2.5 挑选图片	
4.2.6 删除自定义数据集	41
4.3 代码编辑窗口的数据集管理	43
4.3.1 数据集展示	43

Mind Studio 基本操作

4.3.2 Copy Path 功能	
4.3.3 Refresh 功能	44
5 模型管理	46
5.1 简介	46
5.2 Mind Engine 的模型管理	46
5.2.1 模型展示	46
5.2.2 新增自定义模型组件	47
5.2.3 自定义模型加解密	57
5.2.4 新增 Caffe 模型组件	59
5.2.5 模型属性展示	62
5.2.6 模型网络结构查看	63
5.2.7 模型组件删除	66
5.3 代码编辑窗口的模型管理	66
5.3.1 模型展示	66
5.3.2 新增自定义模型组件	68
5.3.3 Copy Path 功能	68
5.3.4 Refresh 功能	69
5.3.5 离线模型导入	69
5.4 参考	71
5.4.1 AIPP 配置说明	71
5.4.2 量化配置	73

1 概述

介绍Mind Studio为开发者提供的服务,整体架构及主要功能。

- 1.1 简介
- 1.2 功能说明

1.1 简介

Mind Studio是一套基于华为NPU(Neural-network Processing Unit)开发的AI全栈开发平台,包括基于芯片的算子开发、以及自定义算子开发,同时还包括网络层的网络移植、优化和分析,另外在业务引擎层提供了一套可视化的AI引擎拖拽式编程服务,极大的降低了AI引擎的开发门槛,全平台通过Web的方式向开发者提供以下4项服务功能。

● 针对算子开发

Mind Studio提供全套的算子开发、支持真实环境运行,支持针对动态调度的异构程序的可视化调试,支持第三方算子开发,极大的降低了基于华为自研NPU的算子开发门槛,提高算子开发效率,有效提升产品竞争力。

● 针对网络层的开发

Mind Studio集成了离线模型转换工具(OMG)、模型量化工具、模型运行 Profiling分析工具和日志分析工具,极大的提升了网络模型移植和分析优化的效率。

● 针对AI引擎开发

Mind Studio提供了AI引擎可视化拖拽式编程以及大量的算法代码自动生成技术,极大的降低了开发者的门槛,并且预置了丰富的算法引擎,如: Resnet18等,大大提高了用户AI算法引擎开发及移植效率。

● 针对应用开发

Mind Studio内部集成了各种工具如Profiler、Compiler等,为用户提供图形化的集成开发环境,通过Mind Studio进行工程管理、编译、调试、仿真、性能分析等全流程开发,从而提高开发效率。

1.2 功能说明

整体架构

Mind Studio整体框架,如图1-1所示。

图 1-1 Mind Studio 整体框架



功能说明

Mind Studio主要提供以下特性功能:

● 基于神经网络芯片 (NPU) 的友好编程界面

针对算子开发人员,Mind Studio基于CCE编程深度定制适配CCE开发,实现深度融合,扩展CCE语言关键字高亮显示、异构混合代码一键式编译等功能。

● 基于神经网络芯片 (NPU) 的图形化调试能力

针对NPU神经网络芯片上的算子加速库开发,Mind Studio提供图形用户接口,实现加速算子在AI Core/AI CPU上的运行状态实时跟踪。

● 离线模型全自动管理

训练好的第三方(Caffe、Tensorflow等)模型可以直接通过Mind Studio导入转换成本系统支持的模型,并可一键式自动生成模型接口,方便用户基于模型接口进行编程。详细功能使用请参见《Mind Studio快速入门》中的"离线模型转换"章节。

● 业务流程编排"0"编码

针对业务流程开发人员,Mind Studio提供基于业务节点的拖拽式编程方式,在Mind Studio上拖拽业务节点并连线,可实现业务编排"0"编码。编排后的编译、运行、结果显示等一站式服务让流程开发更加智能化,整个过程无需编程,简单快捷的通过拖拽和配置完成。用户可以快速上手,可把更多的精力用于理解业务,不用花费额外的学习成本用于了解和使用工具。详细功能使用请参见《Mind Studio快速入门》中的"构建首个机器学习应用"章节。

● TE图形化编程能力

Mind Studio提供了业界首个基于TE(类TVM)的编程开发的集成开发环境,让不同平台下的算子移植更加迅捷,适配NPU速度更快。

● 日志分析

Mind Studio为NPU平台提供覆盖全系统的日志收集与日志分析解决方案,提升运行时算法问题定位效率。统一全系统日志格式,以Web化的形式提供跨平台日志可视化分析能力及运行时诊断能力,提升日志分析系统的易用性。

● 性能分析

Mind Studio有图形界面和命令行两种方式,实现针对Host和Device多节点、多模块异构体系的高效、易用、可灵活扩展的系统化性能分析,以及实现针对NPU Device的性能和功耗的同步分析,满足算法优化对系统性能分析的需求。

● 仿真

提供Caffe开源模型的功能级仿真运行,通过程序调用仿真运行库模拟执行AI核功能。

2 用户管理

Mind Studio提供用户管理功能,用户可以自行修改登录密码。

- 2.1 用户登录
- 2.2 重置密码
- 2.3 修改密码
- 2.4 用户退出
- 2.5 安全日志查询

2.1 用户登录

用户登录界面如图2-1所示。

图 2-1 登录界面



界面参数说明如表2-1所示。

表 2-1 登录界面参数

参数	说明	
"User Name"	登录用户名,默认为 MindStudioAdmin ,不支持修改和新建用户名。	
"Password"	登录密码,初始密码为Huawei123@。密码要求: 密码长度必须在8到16个字符之间。 至少包含一个大写或者小写字母。 至少包含一个数字。 至少包含一个特殊字符(~!@#\$%^&*()=+ \[]{}:;,<>/?)。 说明	
"Login"	登录,输入正确的用户名和密码后,单击该按钮,即可进入Mind Studio界面。	
"Modify"	修改密码,登录Mind Studio后,如果用户想要修改密码,则通过该单击该按钮进行修改。具体的操作请参见 2.3 修 改密码。	

图 2-2 密码过期提示信息

Login Information The password has expired, please change your password!

OK

输入正确的密码后,单击"Login":

- 若用户为首次登录,会弹出密码重置界面,具体的操作请参见**2.2 重置密码**。
- 如果为非首次登录,则会弹出图2-3所示界面,单击"OK"后,直接进入Mind Studio界面。

图 2-3 用户登录信息

Login Information

×

Last login success, login time: 2019-02-21 10:26:09, IP address: 10:142:185:92, number of failed attempts: 0, password validity period: 89 days.

OK

该界面记录上次登录成功与否、登录时间,登录用户的IP地址,尝试失败的次数,以及密码的有效期信息。

- 用户登录成功后,若在30分钟内无任何操作,将提示会话超时,用户被强制 退出,需重新输入密码登录界面。

登录超时的时间可由用户自行配置,您可以在 "~/tools/conf/mind_studio.config"文件中修改 "SessionTimeOut"的值,该参数可以配置的有效范围是 "5分钟~1440分钟",默认为 "30分钟";如果该参数配置为 "-1",表示会话永远不超时;如果设置取值不在有效范围内,则默认使用 "30分钟"。

- 用户登录成功后,在同一客户端同一浏览器可以打开多个Mind Studio界面的页签,但请用户保持在单一页签上使用Mind Studio,多个页签操作可能会导致某些页面功能异常。
- 若当前用户已登录,其他用户用相同的用户名和密码在其他客户端登录Mind Studio,则当前用户会被提示强制退出。
- 若服务器关闭,用户登录会提示如图2-4所示信息。

图 2-4 服务器关闭界面

Login Information

The service is not available!

OK

若出现以上情况,则进入安装时选择的tools路径下(默认路径为"\$HOME/tools")的log文件中,查看mind_log文件,确认是否为Mind Studio后台停止服务,若是,则请重新启动Mind Studio。

2.2 重置密码

用户首次登录Mind Studio,输入正确的用户名和密码后,会弹出重置密码界面,要求用户强制修改密码。界面参数如表2-2所示。

表 2-2 重置密码界面参数

参数	说明
"User Name"	登录用户名,默认为MindStudioAdmin,不支持修 改和新建。
"Old Password"	默认密码。
"New Password"	新密码,不能与"Old Password"内容一致。 密码格式请参见表2-1中的密码要求。 新密码不能与历史3次密码相同。
"Confirm Password"	确认新密码,需要与"New Password"输入内容一致。

密码重置完毕,单击"OK",会弹出重置密码成功界面,单击"OK"后,会重新进入2.1 用户登录所示界面,输入新的密码即可进入Mind Studio界面。

□说明

- 如果5分钟内连续3次执行重置密码失败,重置功能将被锁定,请在30分钟之后再试。
- 重置密码只能执行一次,不可重复执行。

2.3 修改密码

如果用户想要修改密码,则通过单击图2-1中登录界面的"Modify Password",进入密码修改界面,界面参数如表2-3所示。

表 2-3 修改密码界面参数

参数	说明
"User Name"	登录用户名,默认为 MindStudioAdmin ,不支持修 改和新建。
"Old Password"	默认密码。
"New Password"	新密码,不能与"Old Password"内容一致。 密码格式请参见表2-1中的密码要求。 新密码不能与历史使用的3次密码相同,也不能与 重置密码时的Token值相同。
"Confirm Password"	确认新密码,需要与"New Password"输入内容一致。

∭说明

如果5分钟内连续3次执行修改密码失败,修改密码功能将被锁定,请在30分钟之后再试。

密码修改完毕,单击"OK",会弹出修改密码成功界面,单击"OK"后,会重新进入2.1 用户登录所示界面,输入新的密码即可进入Mind Studio界面。

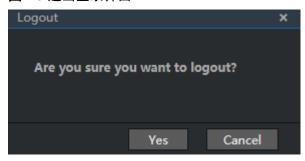
□□说明

若当前用户已登录Mind Studio,其他用户修改密码后,则当前用户会被提示强制退出。

2.4 用户退出

若用户想退出Mind Studio登录界面,依次单击"File>Logout",系统弹出图2-5所示界面,单击"Yes",即可退出登录界面。

图 2-5 退出登录界面



2.5 安全日志查询

安全日志存储在tools路径下(安装时选择的默认路径为"\$HOME/tools/log/login_log")的log文件中,日志会记录每天的登录信息,日志文件记录形式为USR-年月-日.LOG。日志内容形式如图2-6所示。

图 2-6 安全日志



3 工程管理

- 3.1 简介
- 3.2 基本操作

3.1 简介

Mind Studio提供了程序的工程管理功能,可以管理如下多种类型的工程:

- Python工程
- C/C++工程
- HiAI Engine编排工程(Mind工程)
- 基于离线模型开发的C++工程
- Tensor Engine工程

用户可以在上面进行工程管理,具体包括:

- 工程创建/删除
- 工程导入/导出
- 工程开启/关闭
- 文件新建/删除
- 文件/文件夹上传

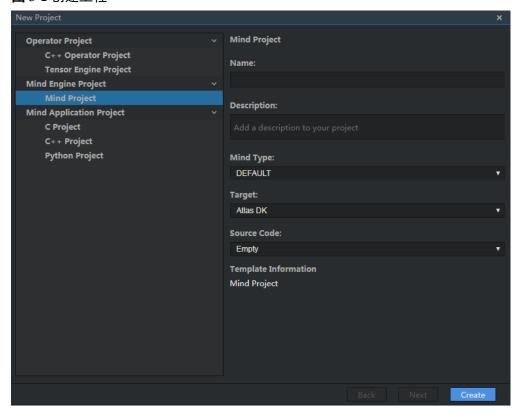
3.2 基本操作

3.2.1 工程创建/删除

● 工程创建:

在Mind Studio的菜单栏中选择"File > New > New Project"创建工程,例如创建Mind工程,在弹出的对话框中单击"Mind Engine Project"条目下的"Mind Project"并输入工程名称,然后单击"Create"创建工程,如图3-1所示,界面参数解释如表3-1所示。

图 3-1 创建工程



∭说明

Mind Studio中新建工程保存在系统根目录下的projects文件夹中。

表 3-1 界面参数说明

参数	参数说明	
Name	工程名称,自行配置。 名字需要为字符串,不包含空格,空格会自动填充为"-"。	
Mind Type	包含如下两种选项: ● DEFAULT: 创建的工程会生成画布,用户可以使用拖拽的方式进行工程编排。 ● CUSTOM: 自定义工程,不会生成画布。	
Target	下拉选择运行环境: ● Local: 仿真环境,仅用于caffe推理引擎。 ● ASIC: 连接EVB、PCIe单板。 ● Atlas DK: 连接开发者板。 说明 如果新建工程为Local仿真工程,则模型必须使用Model中Caffe Models下的模型组件("新增Caffe模型组件"章节请参见5.2.4 新增Caffe模型组件),对应预处理选择ImagePreProcessPillow节点,推理引擎对应必须为CaffeInferenceEngine。	

参数	参数说明
Source Code	代码来源:
	● Empty: 选择该项,表示工程非外部导入。
	● Local(Web Client): 从本地Windows中导入源文件,选择该项,下方出现文件上传输入框。
	● Local(Web Server): 从后端服务器导入源文件,选择该项,则下方出现输入框填写Mind Studio服务器端代码路径。

● 工程删除:

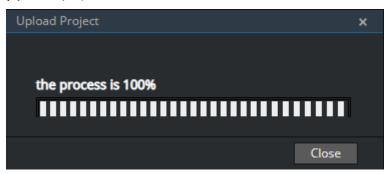
选中工程根目录, 鼠标右键, 在弹出的菜单中单击"Delete"即可。

3.2.2 工程导入/导出

● 工程导入:

在Mind Studio的"File"菜单选择"Upload Project"弹出导入工程选择对话框,选择要导入工程的zip文件上传,上传成功后出现进度提示,如图3-2所示。

图 3-2 工程导入



● 工程导出:

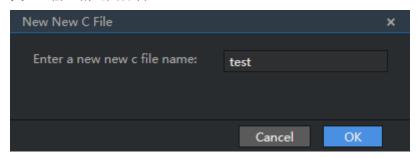
在Mind Studio的 "File"菜单选择Download Project,Project会以zip压缩包的形式下载到本地用户download目录下,对于Mind工程中加入的自定义dataset和自定义model,其中的数据也同样会随工程一起导出,保存在zip压缩包中的MyDataset和MyModel文件夹中(在重新导入该zip压缩包时,MyDataset和MyModel中的数据也会被重新加载到对应的数据集和模型目录中)。

3.2.3 新建/删除文件

● 新建文件:

在Projects Explorer视图中,单击鼠标右键选择"New"(或者在File菜单中选择New),在弹出的子菜单中选择需要创建的文件类型,在打开的对话框中输入文件名(不需要输入文件类型对应的后缀名),如图3-3所示。

图 3-3 输入新建文件名



● 删除文件:

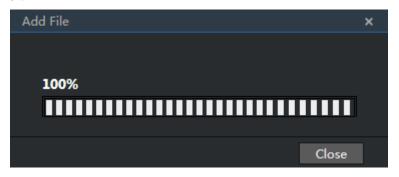
选中文件, 鼠标右键, 在弹出的菜单中单击"Delete"即可。

3.2.4 文件/文件夹上传

● 文件上传:

在Projects Explorer视图中选中一个文件夹,依次单击"File > Add File"打开文件上传对话框,选择本地文件后上传,上传成功后,如图3-4所示。

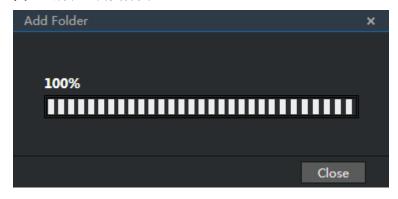
图 3-4 成功上传文件



● 文件夹上传:

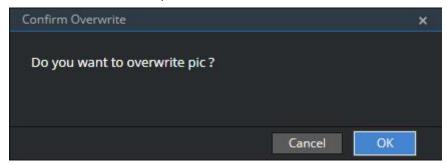
在Projects Explorer视图中选中一个文件夹,依次单击"File > Add Folder"打开文件夹上传对话框,选择本地文件夹后上传,上传成功后如图3-5所示。

图 3-5 成功上传文件夹



如果工程中已经存在同名文件/文件夹,则会弹出对话框提示,如图3-6所示。

图 3-6 提示覆盖同名文件/文件夹



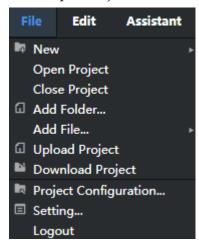
∭说明

请确保上传的文件夹根目录中有文件、空的文件夹不会被上传。

3.2.5 打开项目

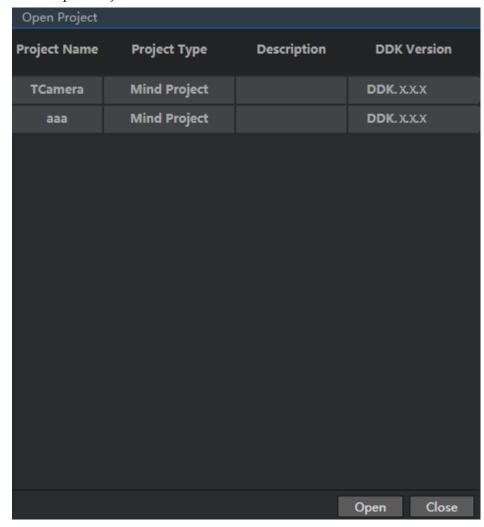
在已经有Project关闭的情况下,单击File菜单中的Open Project子菜单,如图3-7所示。

图 3-7 Open Project 子菜单



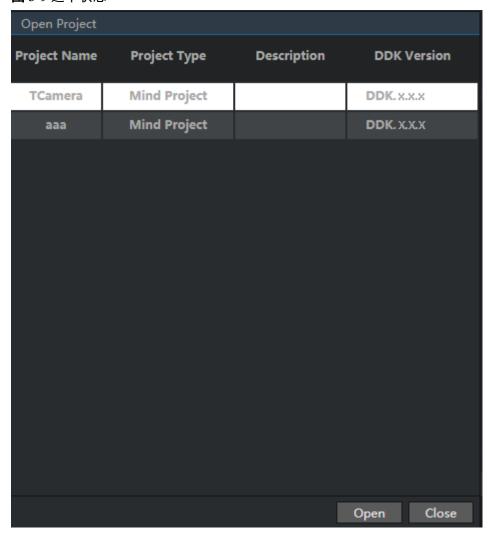
单击后弹出Open Project界面,如图3-8所示。

图 3-8 Open Project 界面



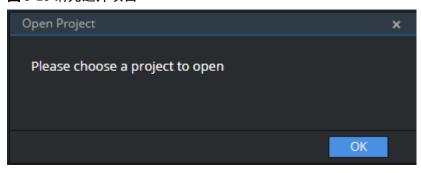
选中一个项目并打开,如图3-9所示。

图 3-9 选中状态



若未选中项目,则会提示先选择项目,并返回刚才的界面,如图3-10所示。

图 3-10 请先选择项目

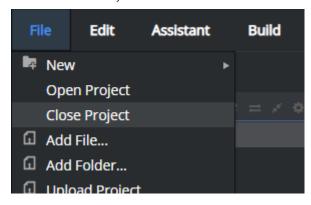


成功后,打开的项目会在Projects Explorer中成功显示。

3.2.6 关闭项目

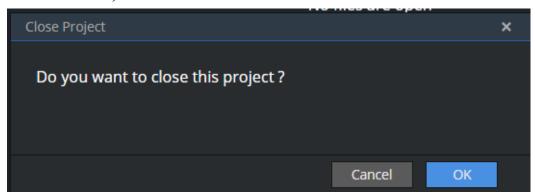
在Projects Explorer中选中一个Project,然后单击File菜单中的"Close Project"子菜单,如图3-11所示。

图 3-11 Close Project 子菜单



之后询问是否需要关闭该项目,如图3-12所示。

图 3-12 Close Project 界面



若选择"OK",则将项目进行关闭,在Projects Explorer内找不到该项目,该项目对应打开的file文件也全部关闭。

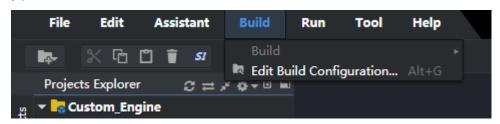
3.2.7 工程编译

□ 说明

针对C/C++工程、Python工程,不支持编译、运行、debug和Profiling功能。

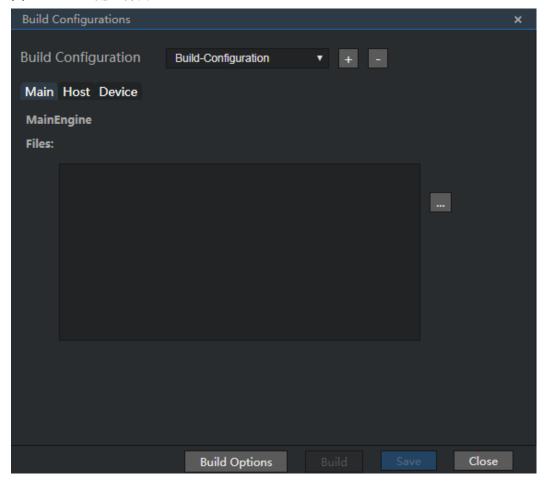
在Projects Explorer中选中一个项目,然后依次单击"Build > Edit Build Configuration...",如图3-13所示。

图 3-13 Build 菜单



之后弹出Build配置界面,如图3-14所示。

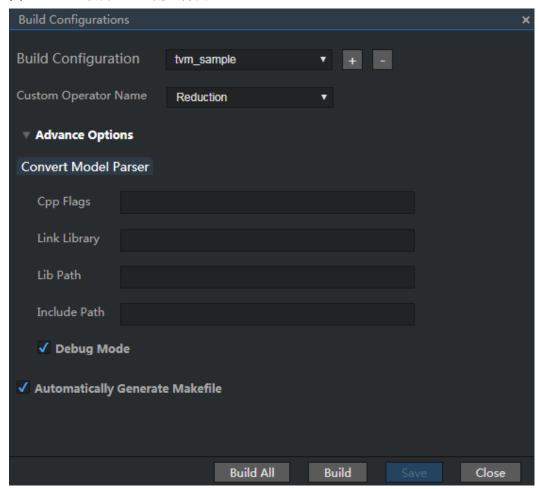
图 3-14 Build 配置界面



在配置界面中可以单击"+"新增一个配置,也可以在下拉列表中选择其中一个配置进行删除。下拉列表中选中不同的配置,Build配置界面随之改变。

如果选中的项目为tvm项目(创建的工程为Tensor Engine Project)时出现的Build配置,如图3-15所示。

图 3-15 tvm 项目 Build 配置界面



单击 "Advance Options"会显示"Convert Model Parser"区域及其参数,界面参数解释如表3-2所示。

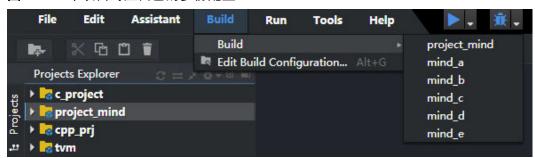
表 3-2 tvm 项目 Build 配置界面参数

参数	说明	
Build Configuration	工程名。	
Custom Operator Name	需要编译的算子。	
Cpp Flags	编译选项,用户配置。	
Link Library	引用的链接库,用户可配置。 链接的动态库,对应编译命令中-l参数的值。	
Lib Path	链接库路径,目前使用缺省路径,用户可增加配置。 动态库查找路径,对应编译命令中-L参数的值。	

参数	说明
Include Path	头文件路径,目前使用缺省路径,用户可增加配置。 头文件查找路径,对应编译命令中-I参数的 值。
Debug Mode	是否开启debug模式,对应-g 编译选项。
Automatically Generate Makefile	是否自动生成Makefile文件。

在Projects Explorer中选中其中一个项目,之后依次单击"Build > Build",会显示选中项目的多份配置,如图3-16所示。

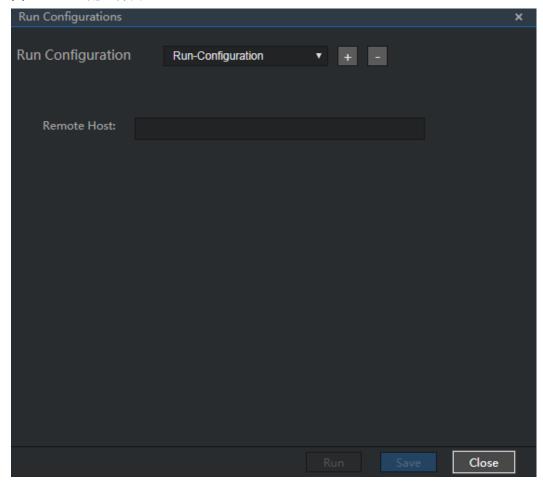
图 3-16 一个项目对应自己的多份配置



3.2.8 工程运行

在Projects Explorer中选中一个项目,然后依次单击 "Run > Edit Run Configuration...",弹出Run配置界面,如图3-17所示。

图 3-17 Run 配置界面



在配置界面中单击"+",新增一个配置,也可以在下拉列表中选择其中一个配置进行删除,下拉列表中选择不同的配置,Build配置界面随之改变。

在Projects Explorer中选中一个项目,在Run菜单中选择Run子菜单,只会显示选中项目的配置。

3.2.9 支持的文件格式

目前Mind Studio支持的所有文件格式请参见zip包中resource文件夹下的file format.zip。

4 数据集管理

- 4.1 简介
- 4.2 Mind Engine的数据集管理
- 4.3 代码编辑窗口的数据集管理

4.1 简介

数据集分为内置数据集和自定义数据集,其中内置数据集用户可以直接拖拽使用,自定义数据集需要用户手动导入。自定义数据集属性信息存放在mongodb中,用户导入的文件存放在文件系统中。

● 内置数据集

由Mind Studio开发者提供的内置数据集,用户可以在自己创建的Mind工程中使用内置数据集。内置数据集在创建工作空间时就存在了,用户可以使用,但不能对其进行增加,删除和修改。

当前已经设置的内置数据集:

MnistDataset: 最常用的合理性检验数据集,由黑白手写数字图像组成,图像大小为28x28,数字居中显示。MNIST是一项比较简单的任务,通过MNIST测试不一定表明模型本身能有效运作。

● 自定义数据集

通过增加自定义数据集功能,可以将图片的集合保存为自定义数据集,供用户后续使用。用户可以自行增加自定义数据集,新创建的工作空间自定义数据集是空的。自定义数据集可增加或删除,但不可修改。

自定义数据集可以通过本地文件和本地文件夹来获取图片来源。

4.2 Mind Engine 的数据集管理

4.2.1 数据集展示

双击Mind Engine类型工程中.mind文件进入Engine编排窗口。

右侧的tool界面中Datasets为数据集管理区域,分为2个子目录,如图4-1所示。

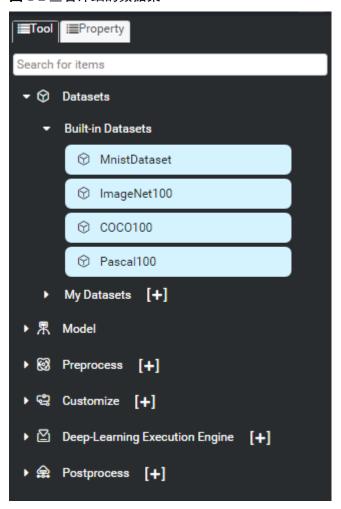
- BuiltIn Datasets: 内置数据集。
- My Datasets: 自定义数据集。

图 4-1 数据集管理区域



单击 能展开该目录,看到目录具体内容,如**图4-2**所示。

图 4-2 查看详细的数据集



4.2.2 数据集导入

不同数据类型的导入

您可以根据不同的数据类型导入不同的数据集,数据类型分类以及区别如表4-1所示。

表 4-1 数据集介绍

数据集	输入图片后缀	是否带label文件
Image	nv12(YUV420SP格式)、jpeg、png、 jpg、bmp、JPEG、JPG、PNG、 BMP	否
Raw	bin(BGR float格式)	否
ImageNet	jpeg、JPEG	是
COCO	jpg	是
PASCAL	jpg	是
Camera	N/A	否
MIC	N/A	否

∭说明

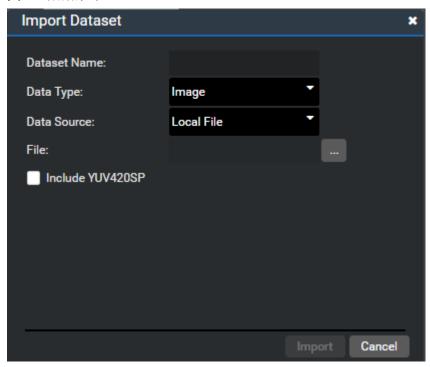
限制:图片的宽和高的范围为[16,4096]。

单击My Datasets右侧的 并,弹出"Import Dataset"数据集导入窗口,如图4-3和图4-4 所示。

图 4-3 数据集导入



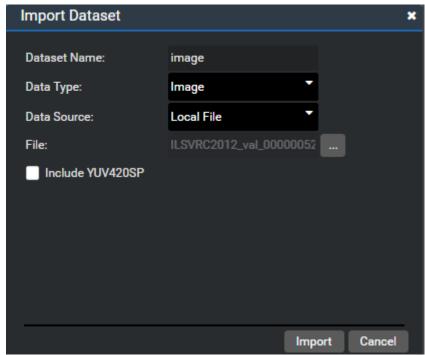
图 4-4 数据集导入 2



● Image数据集导入

- 导入的图片中不包含YUV420SP格式的图片,如图4-5所示。
 - i. "Data Type"中选择Image类型。
 - ii. "Dataset Name"中输入数据集的名称。
 - iii. "Data Source"中选择数据集要导入的方式,详情请参见不同数据源的导入。
 - iv. "File"中,单击 选择要导入的图片。
 - v. 当Dataset Name和File都已设置,且不包含YUV420SP格式的图片,import 按钮可用,单击"import"导入数据集,如图4-5所示。

图 4-5 Image 数据集导入过程 1



□ 说明

如果Data Source选择Local Folder,选择的目录下必须都是图片。

- 导入的图片中包含YUV420SP格式的图片,如图4-6、图4-7所示。
 - i. 勾选Include YUV420SP。
 - ii. 在"Width"和"Height"中分别输入图片的宽和高。

□ 说明

数据集的宽和高取值为整型。

iii. 当"Dataset Name"和"File"都已设置,且包含YUV420SP格式的图片,并且宽和高都已经输入正确,Import按钮可用,单击"Import"。

图 4-6 Image 数据集导入过程 2

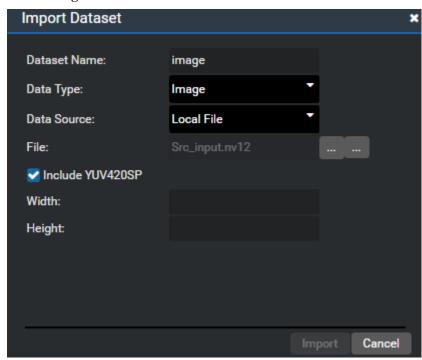
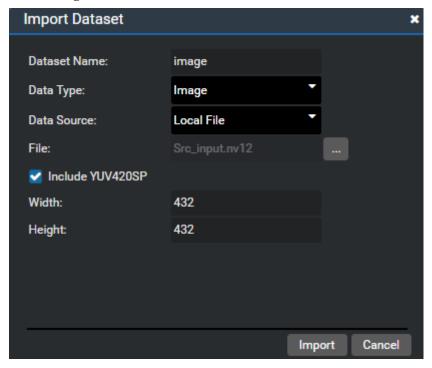


图 4-7 Image 数据集导入过程 3



□□说明

同一个数据集的所有YUV420SP的图片中Width和Height必须相同。

- Raw数据集导入,如图4-8、图4-9所示。
 - a. "Data Type"中选择"Raw"类型。

- b. "Dataset Name"中输入数据集的名称"image"。
- c. "Data Source"中选择数据集要导入的方式"Local File",详情请参见不同数据源的导入。
- d. 输入该数据集的宽和高。

□说明

数据集的宽和高取值为整型。

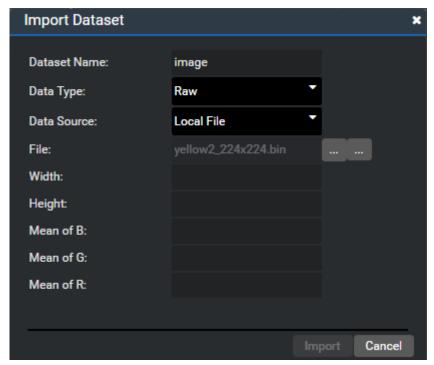
e. 输入该数据集的均值,均值范围为[0,255]。

□说明

均值类型为浮点型,且小数点后最多支持4位。

f. 当Dataset Name和File都已设置,且宽高和均值都已设置,Import按钮可用, 单击"Import"导入数据集。

图 4-8 Raw 数集导入过程 1



Import Dataset Dataset Name: image Data Type: Raw Data Source: Local File yellow2_224x224.bin File: Width: 224 Height: 224 Mean of B: 102.9801 Mean of G: 115.9465 Mean of R: 122.7717 Cancel **Import**

图 4-9 Raw 数集导入过程 2

□说明

同一个Raw数据集里每个文件的Width、Height、均值必须相同。

● ImageNet数据集导入

- Use Ground Truth和Use Label都选择,如图4-10所示。
 - i. "Data Type"中选择"ImageNet"类型。
 - ii. "Dataset Name"中输入数据集的名称"imagenet"。
 - iii. "Data Source"中选择数据集要导入的方式"Local File",详情请参见不同数据源的导入。
 - iv. 默认使用Ground Truth,可不使用Use Ground Truth。此处使用,选择Ground Truth File。
 - v. 默认使用Label,可不使用Use Label。此处使用,选择Label File。

□□说明

Ground Truth File为.csv文件,表明图片对应的是什么数字; Label File为是json文件,表明数字对应的物体是什么。

vi. 当Dataset Name和File都已设置,且选择Ground Truth File和Label File,Import按钮可用,单击"Import"导入数据集。

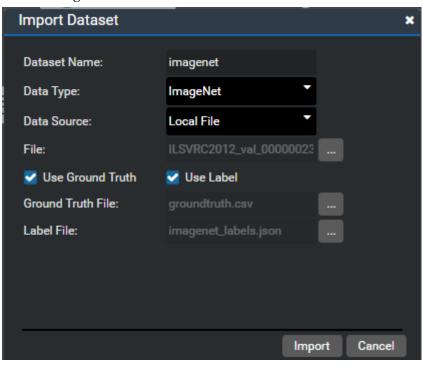


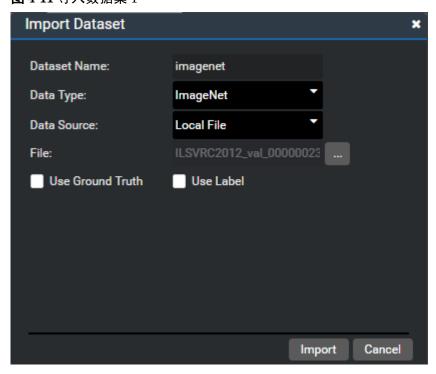
图 4-10 ImageNet 数据集导入过程

□说明

如果Data Source选择Local Folder,选择的目录下必须都是图片。

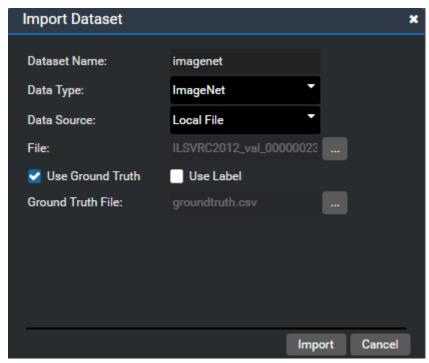
Use Ground Truth和Use Label都不选择,如图4-11所示。 当Dataset Name和File都已设置,Import按钮可用,单击"Import"导入数据集。

图 4-11 导入数据集 1



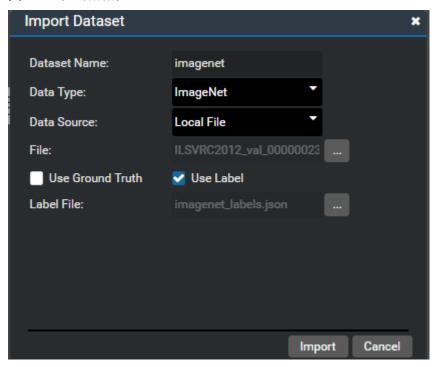
- **只选择Use Ground Truth**,如图4-12所示。
 - i. 选择 "Use Ground Truth", 去勾选"Use Label"。
 - ii. "Ground Truth File"中导入.csv的文件。
 - iii. 当Dataset Name和File都已设置,且Ground Truth File中文件已导入,Import按钮可用,单击"Import"导入数据集。

图 4-12 导入数据集 2



- **只选择Use Label**,如<mark>图4-13</mark>所示。
 - i. 选择**"Use Label"**,不勾选"Use Ground Truth"。
 - ii. "Label File"中导入.json的文件。
 - iii. 当Dataset Name和File都已设置,且Label File中文件已导入,Import按钮可用,单击"Import"导入数据集。

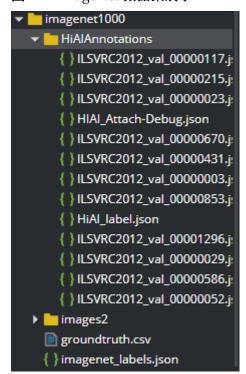
图 4-13 导入数据集 3



□ 说明

如果选择Use Ground Truth或Use Label,会在数据集目录下生成一个目录 HiAIAnnotations,在该目录下存放每张图片的注解文件;且使用label时,还会在该目录下生成对应的标签字典文件HiAI_label.json文件,如图4-14所示。

图 4-14 ImageNet 数据集展示



● **COCO数据集导入**,如**图4-15**所示。

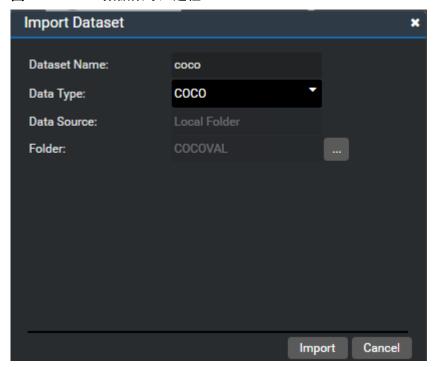
- a. "Data Type"中选择"COCO"类型。Data Source只能是Folder。
- b. "Dataset Name"中输入数据集的名称,例如"coco"。
- c. 单击 "Folder"后面的 选择一个目录。

□ 说明

该目录下必须包含Annotations和Images目录,且Annotations目录里面必须包含标准的instances的json文件,Images目录下都是jpg图片。

d. 当Dataset Name和Folder都已设置,Import按钮可用,单击"Import"导入数据集。

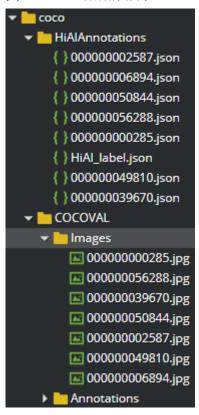
图 4-15 COCO 数据集导入过程



∭说明

导入成功后,会在数据集目录下生成一个目录HiAIAnnotations,该目录存放每张图片的注解文件和label文件HiAI_label.json,如图4-16所示。

图 4-16 coco 数据集展示



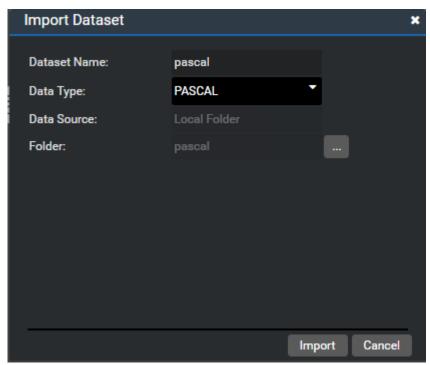
- PASCAL数据集导入,如图4-17所示。
 - a. "Data Type"中选择"PASCAL"类型。Data Source只能是Folder。
 - b. "Dataset Name"中输入数据集的名称"pascal"。
 - c. 单击 "Folder"后面的 选择一个目录。

□ 说明

该目录下必须包含Annotations和JPEGImages目录,且Annotations目录里面包含每张图片的xml文件,JPEGImages目录下都是jpg图片。

d. 当Dataset Name和Folder都已设置,Import按钮可用,单击"Import"导入数据集。

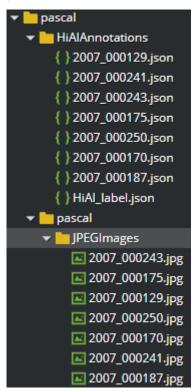
图 4-17 PASCAL 数据集导入过程



∭说明

会在数据集目录下生成一个目录HiAIAnnotations,该目录存放每张图片的注解文件和 label文件HiAI label.json,如图4-18所示。

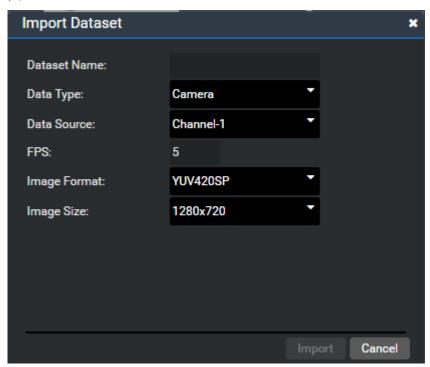
图 4-18 PASCAL 数据集展示



● Camera类型,如图4-19所示。

- a. "Data Type"中选择Camera类型。
- b. "Dataset Name"中输入Camera数据集的名称。
- c. "Data Source"中选择Camera的通道,当前的通道包括Channel-1、Channel-2。
- d. "FPS"中设置Camera的帧率,当前fps的设置范围为[1,20]。
- e. "Image Format"中设置采集的图片格式,当前仅支持YUV420SP。
- f. "Image Size"中设置采集的图片大小, 当前仅支持1280*720。

图 4-19 Camera 数据集导入过程

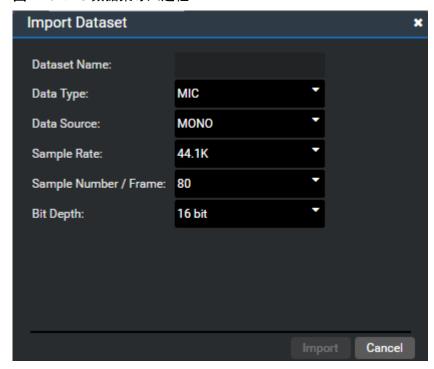


□□说明

Camera数据集当前仅支持Atlas DK开发板场景。

- **MIC类型**,如**图4-20**所示。
 - a. "Data Type"中选择MIC类型。
 - b. "Dataset Name"中输入MIC数据集的名称。
 - c. "Data Source"中选择MIC的通道,当前的通道包括MONO(单声道)、 STEREO(立体声)。
 - d. "Sample Rate"中设置MIC采样率,当前采样率的设置选项包括: 8K、11.025K、12K、16K、22.05K、32K、44.1K、48K、64K、96K。
 - e. "Sample Number/Frame"中设置采集的每一帧数据包括多少个样本,当前每一帧的样本包括: 80、160、240、480、1024、2048。
 - f. "Bit Depth"中设置每个样本的位数。

图 4-20 MIC 数据集导入过程



∭说明

- MIC数据集当前仅支持Atlas DK开发板场景。
- 由于硬件限制,每一帧数据之间的间隔要>=10ms,要求Sample Number/Frame>=Sample Rate/100。

不同数据源的导入

- 通过Local File导入数据集,如图4-21所示。
 - a. "Data Source"下拉框选择"Local File"。
 - b. 单击 "File"后的 选择一个图片。

图 4-21 Local File 导入过程



- 通过Local Folder导入数据集,如图4-22所示。
 - a. "Data Source"下拉框选择"Local Folder"。
 - b. 单击"File"后的 选择一个目录。

图 4-22 Local Folder 导入过程



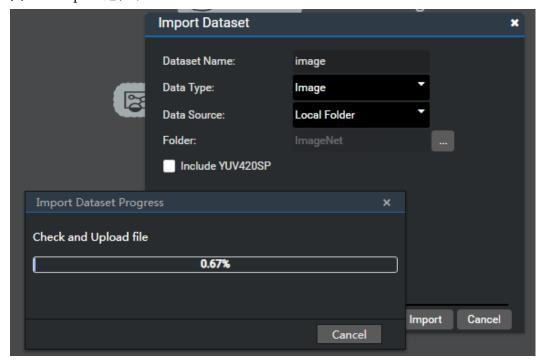
∭说明

规格为5万个文件,如果选择的文件夹文件个数超过5万,可能会导致浏览器界面卡住,出现"页面无响应"信息,此时如果单击"等待",Folder的输入框也可能会加载成功,出现文件夹名。

后续处理

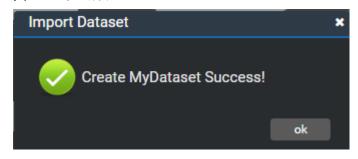
Import过程中的进度条,表示正在进行导入,如图4-23所示。

图 4-23 import 过程中



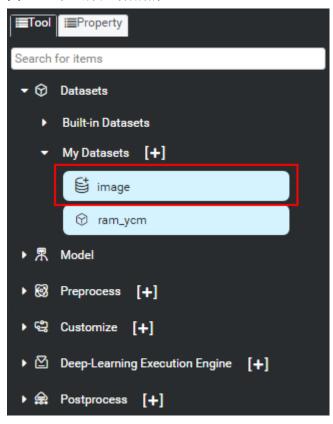
当出现如图4-24所示窗口,表示创建成功。

图 4-24 导入成功



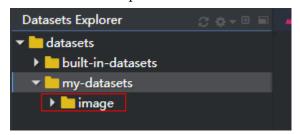
展开工具栏的My Datasets,会增加新增的Dataset Name组件,表明导入数据集成功,可以拖拉该组件使用,如图4-25所示。

图 4-25 导入成功的数据集



同时展开Dataset Explorer区域的my-datasets目录,点击型进行刷新,my-datasets目录下会增加新的Dataset Name目录,如图4-26所示。

图 4-26 Datasets Explorer 中显示新增数据集



4.2.3 数据集属性展示

业务节点的属性配置是运行该节点所需的参数。

右边区域框内的"property"栏为属性设置栏,选中Dataset节点即可对其属性进行设置,如图4-27、图4-28所示。

图 4-27 内置数据集 COCO100 属性配置

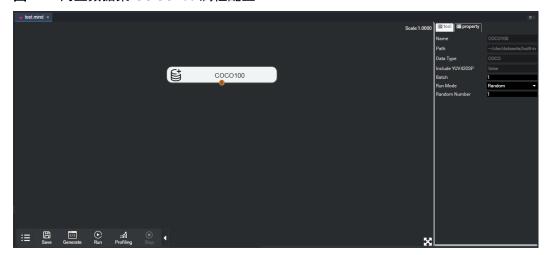


图 4-28 自定义数据集 Raw 属性配置



节点的属性信息如表4-2所示。

表 4-2 节点属性信息

属性名	说明	
Name	Dataset名,Import Dataset时设置。	
Path	Dataset存放在Mind Studio上的路径,Import Dataset时设置。	
Data Type	数据类型,Import Dataset时选择。	
IncludeYUV420S P	Dataset是否包含YUV420SP的格式,Import Dataset时设置。	
Height	仅在包含YUV420SP或Data Type为Raw时显示对应的高,Import Dataset时设置。取值为整型。	
Width	仅在包含YUV420SP或Data Type为Raw时显示对应的宽,Import Dataset时设置。取值为整型。	
Mean of B	仅在Data Type为Raw时显示B的均值,Import Dataset时设置。取值为浮点型,小数点后最多支持4位。	

属性名	说明	
Mean of G	仅在Data Type为Raw时显示G的均值,Import Dataset时设置。取值为浮点型,小数点后最多支持4位。	
Mean of R	仅在Data Type为Raw时显示R的均值,Import Dataset时设置。取值为浮点型,小数点后最多支持4位。	
Batch	一次处理几张图片,可由用户编辑。	
Run Mode	运行模式: All: 跑所有图片。Specify: 处理用户挑选图片。 Random: 处理随机选择的Random Number张图片,可由用户编辑。	
Random Number	仅在Run Mode为"Random"时显示,设置的值范围在[1,数据集图片个数),如果图片个数为1,仅设置为1才有效,可由用户编辑。	

4.2.4 生成 cpp 文件

Dataset组件拖拽后会自动生成.cpp、.h文件,该.cpp或.h文件将会显示在左边目录下,以供用户全流程编排使用。如图4-29所示。

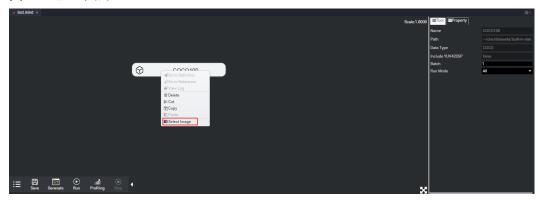
图 4-29 生成 cpp 文件



4.2.5 挑选图片

当一个数据集的图片过多,用户不想全部图片都运行,只想运行其中几张,右击 Dataset组件,选择"Select Images",如图4-30所示。

图 4-30 挑选图片



在弹出界面中挑选几张图片,勾选图片的右上角,单击"select",如图4-31所示。

图 4-31 单击 select



该数据集的Run Mode属性就变为"Specify",表示仅运行选择的图片。

4.2.6 删除自定义数据集

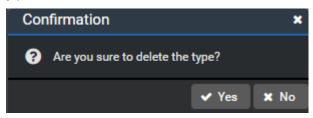
步骤1 右键单击My Datasets下的某个数据集,出现delete按钮,单击"delete",如**图4-32**所示。

图 4-32 删除自定义数据集



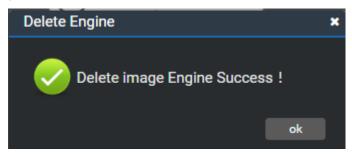
步骤2 在弹出页面中选择Yes,如图4-33所示。

图 4-33 删除自定义数据集弹出页面



步骤3 当出现以下窗口时,表示数据集删除成功,如<mark>图4-34</mark>所示。

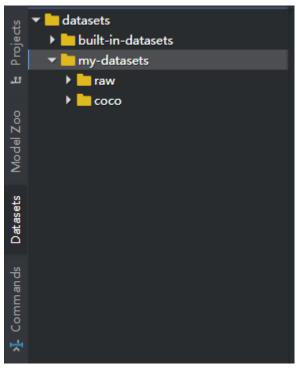
图 4-34 删除自定义数据集成功



步骤4 单击ok, 关闭该窗口。

步骤5 查看Datasets Explorer区域,展开my-datasets,点击 进行刷新,发现数据集image对应的目录已删除,如图4-35所示。

图 4-35 查看 Datasets Explorer



----结束

4.3 代码编辑窗口的数据集管理

4.3.1 数据集展示

单击左侧菜单栏中的Datasets会出现Datasets Explorer,该Explorer展示了datasets的目录信息,分为两个子目录,如图4-36所示。

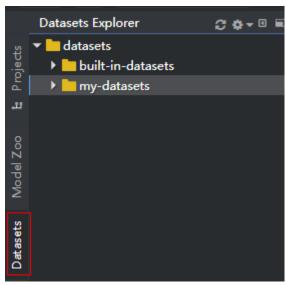
- built-in-datasets,即内置数据集。
- my-datasets,即自定义数据集。

□□说明

my-datasets下的目录为受控目录,不要在my-datasets下存放不是从Mind Studio导入的数据集数据。

- 1. 在后台服务重启时,会做数据一致性校验,如果my-datasets下目录不在mongodb中,该 目录下的数据会被自动删除。
- 2. 如果用户从后台在my-datasets目录下自行创建了一个目录,再从Mind Studio导入与该目录同名的数据集,那么Mind Studio在数据导入前,会先删除用户创建的该目录。

图 4-36 Datasets Explorer

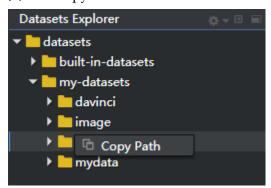


单击 可以看到目录详细内容。

4.3.2 Copy Path 功能

选择一个Dataset的目录或文件,单击鼠标右键,会出现Copy Path,单击Copy Path,如图4-37所示,会把目录或文件的后台服务器路径复制到剪切板,以供用户使用该Dataset。

图 4-37 Copy Path



可以在安装Mind Studio后台服务器打开datasets目录,如图4-38所示。

图 4-38 进入自定义 image 数据集目录示例

ascend@szvphicpra61963:~\$ cd /home/ascend/tools/che/datasets/my-datasets/imagenet ascend@szvphicpra61963:~/tools/che/datasets/my-datasets/imagenet\$ ■

4.3.3 Refresh 功能

新增我的数据集,会在Datasets中生成新的文件夹(文件)。选择一个文件夹,如图 4-39,单击Refresh图标,会刷新该路径下一层的信息(如果展开多层,也只刷新一层,因为有可能目录中文件过多,造成性能问题),刷新后如图4-40。

图 4-39 Refresh 前

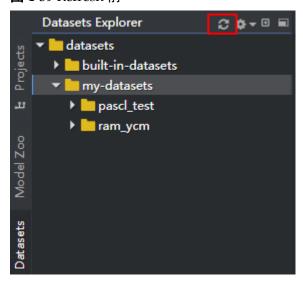
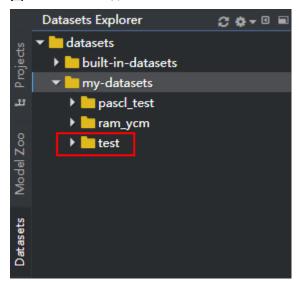


图 4-40 Refresh 后



□ 说明

基于性能原因考虑,暂不支持通过展开折叠文件夹的方式刷新目录,如果要刷新目录,只能通过 单击
■实现。

5 模型管理

- 5.1 简介
- 5.2 Mind Engine的模型管理
- 5.3 代码编辑窗口的模型管理
- 5.4 参考

5.1 简介

模型分为内置模型、自定义模型和Caffe模型。

● 内置模型

内置模型由Mind Studio开发者提供,在创建工作空间时就已经存在,用户可以在自己创建的mind工程中使用内置模型,但不能对其进行增加,删除和修改。 当前已经设置的内置模型: Resnet18

● 自定义模型

通过增加自定义模型功能,可以将Caffe/Tensorflow等模型转换为自定义离线模型,供用户后续使用。用户可以自行增加自定义模型,刚创建的工作空间自定义模型是空的。自定义模型可以通过模型转换或者新增模型功能增加。

● Caffe模型

通过编排界面的新增Caffe模型,可以将Caffe模型加入到model-zoo中,刚创建的工作空间Caffe模型是空的。

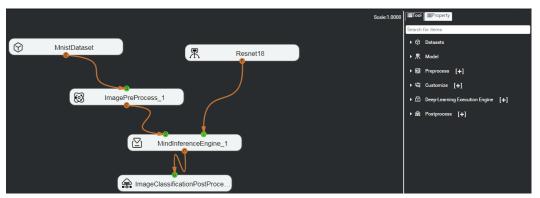
5.2 Mind Engine 的模型管理

5.2.1 模型展示

步骤1 双击Mind Engine类型工程中.mind文件进入Engine编排窗口。

步骤2 tool界面的Model区域为网络模型管理区域,分为3个子目录,如图5-1所示。

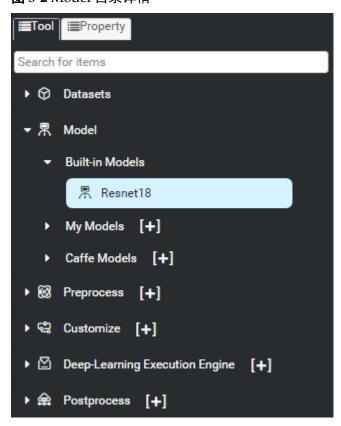
图 5-1 Model 区域展示



- BuiltIn Models,即内置模型。
- My Models,即自定义模型。
- Caffe Models,即Caffe模型。

步骤3 单击▶展开目录,可以看到具体内容,如图5-2所示。

图 5-2 Model 目录详情



----结束

5.2.2 新增自定义模型组件

自定义模型组件添加有两种方式:

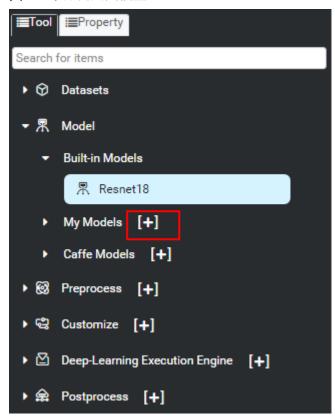
- 在.mind文件的编排窗口中,通过"tool>My Models"新增的方式生成。
- 选中工程后,通过右击选择 "Convert Model..."或者界面选择 "Tools > Convert Model..."模型转化的方式生成。

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

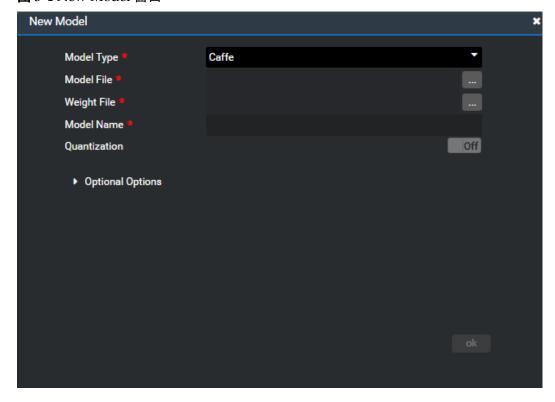
步骤1 单击My Models右侧的 ,添加自定义模型组件,如图5-3所示。

图 5-3 添加自定义模型



步骤2 弹出New Model编辑窗口,如图5-4所示。

图 5-4 New Model 窗口



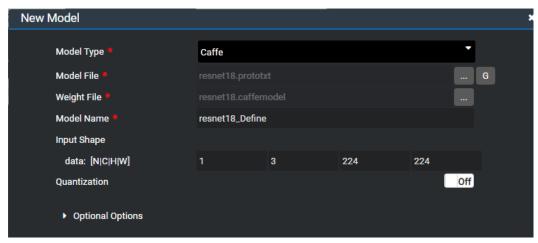
"Model type"可选择Caffe,Tensorflow或OfflineModel:

- 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5-5所示。

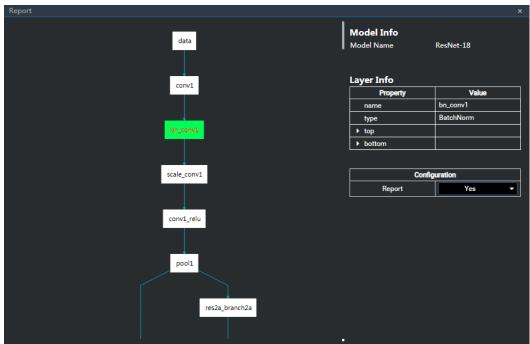
图 5-5 选择模型文件



- "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 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5-6所示。

图 5-6 模型网络结构



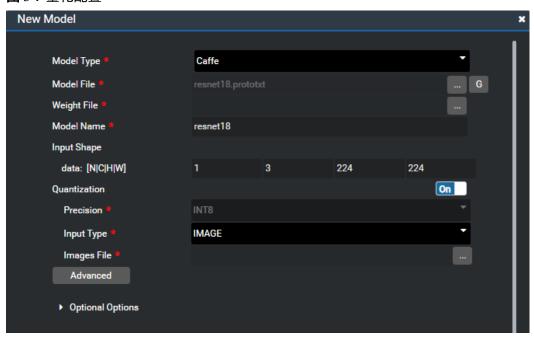
步骤5 打开"Quantization"开关,可以进行量化设置。

"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。

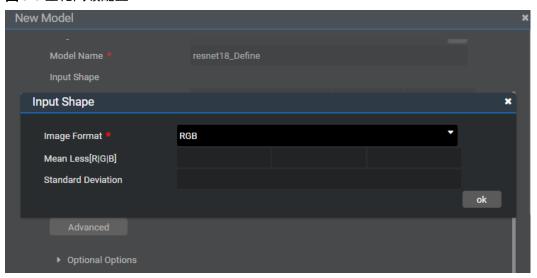
建议选择不超过50张图片进行量化,否则有可能因为量化时间太长导致进程超时(3小时)。如**图5-7**所示。

图 5-7 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5-8所示。

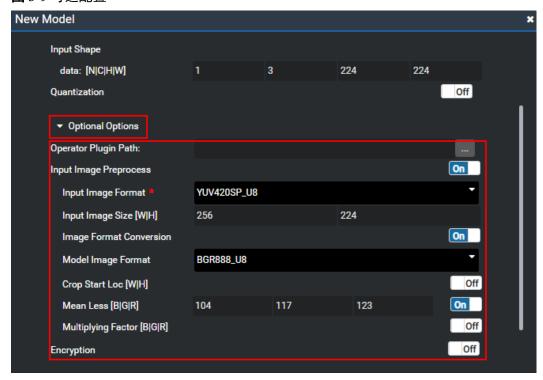
图 5-8 量化高级配置



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**5.4.2** 量化配置。

步骤7 单击"Optional Options",可进行更多可选配置,如图5-9所示。

图 5-9 可选配置



配置项说明如表5-1所示。

表 5-1 Optional Options 参数说明

参数名称	参数描述
Operator Plugin Path	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 5.2.3 自定义模型加解密 。

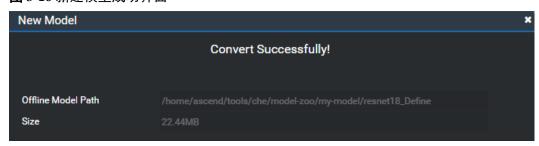
∭说明

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5-10所示。

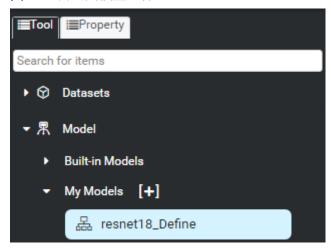
图 5-10 新建模型成功界面



如果失败并且弹出Error Report界面,请参见《Ascend 310 Mind Studio快速入门》中的 *离线模型转换*章节。

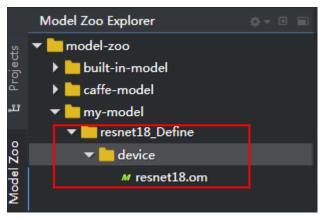
新增模型成功后,会在自定义模型组件中,可直接拖拽用于后续编排,如图5-11所示。

图 5-11 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MS-12所示。

图 5-12 转化后的模型文件



四湖

resnet18.om即为转换生成的模型文件。

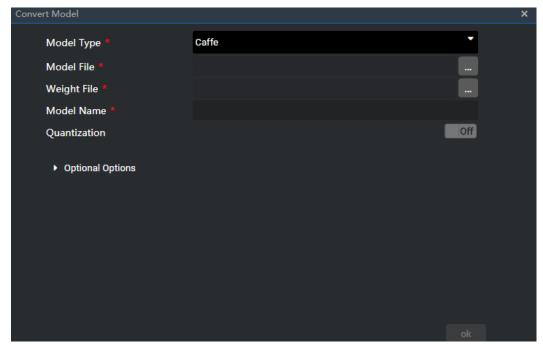
----结束

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。

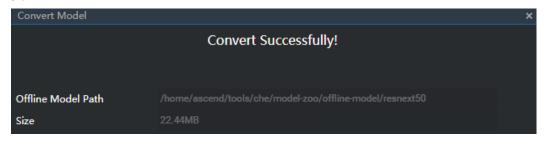
步骤2 右击选择 "Convert Model..."或者界面选择 "Tools > Convert Model..."。 弹出模型转化配置界面,如图5-13所示。

图 5-13 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5-14所示。

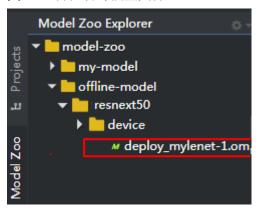
图 5-14 模型转换成功界面



如果失败并且弹出Error Report界面,请参考《Ascend 310 Mind Studio快速入门》中的 离线模型转换章节。

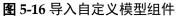
步骤6 模型转化成功后可以在"model-zoo > offline-model"中看到自定义的模型组件,如图 5-15所示。

图 5-15 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5-2 所示,导入后的界面如图5-16所示。



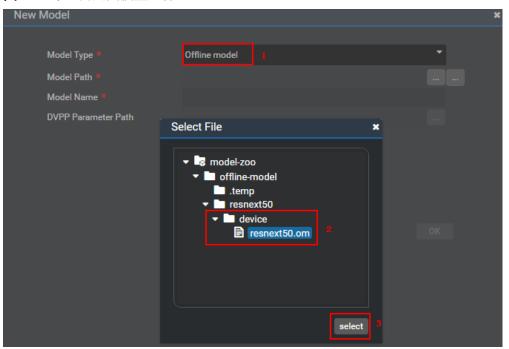


表 5-2 离线模型导入界面参数

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

5.2.3 自定义模型加解密

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。

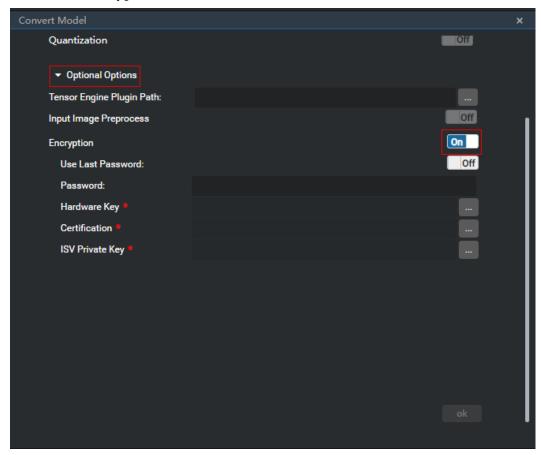
加密的模型仅只支持在"Target"为"ASIC"或"Atlas DK"的Mind工程中运行。

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5-17所示。





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5-18所示。

图 5-18 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5-19所示。

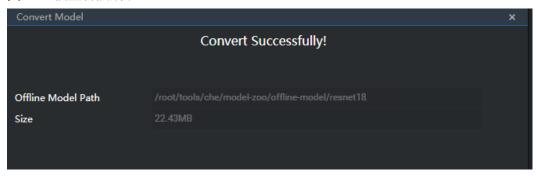
图 5-19 上传密钥、证书以及私钥



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5-20所示。

图 5-20 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

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5-21所示。

图 5-21 模型属性展示页面



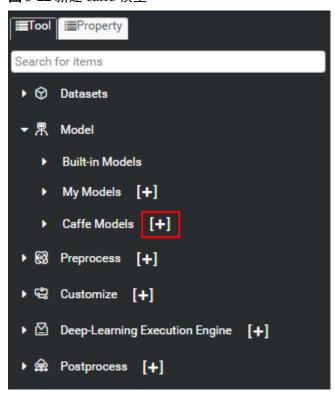
步骤3 打开Decryption开关,选择Passcode文件,解密后,可进行正常开发。

----结束

5.2.4 新增 Caffe 模型组件

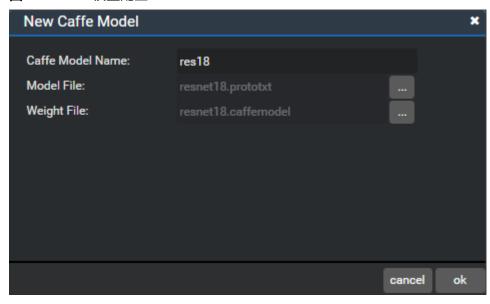
步骤1 单击Caffe Models的 ,新建模型,如图5-22所示。

图 5-22 新建 caffe 模型



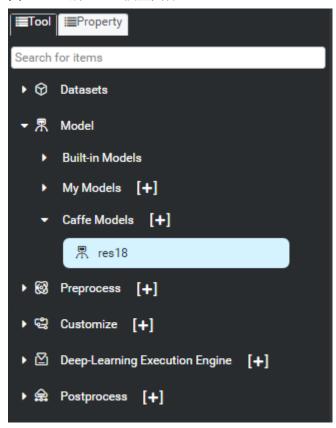
步骤2 在弹出的新建模型窗口,输入Caffe模型名称,选择模型文件和权重文件,单击 "ok",如图5-23所示。

图 5-23 Caffe 模型配置



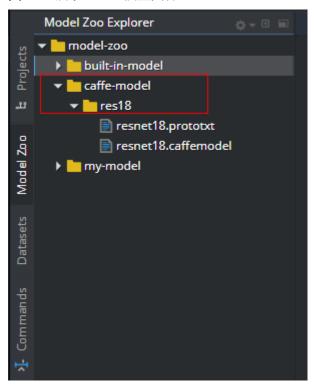
步骤3 在左边tool页签的Caffe Models下生成了一个新的Caffe模型组件,如图5-24所示。Caffe 模型组件的使用方式和My Model基本相同,拖拉到画布中,连接Caffe推理引擎,可以运行Caffe模型推理流程。

图 5-24 生成 Caffe 模型文件



步骤4 新建成功后,同步在model-zoo的caffe-model下展示caffe文件,如图5-25所示。

图 5-25 展示 Caffe 模型文件



----结束

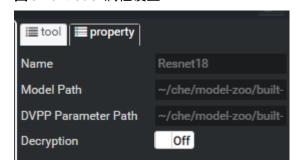
5.2.5 模型属性展示

业务节点的属性配置是运行该节点所需的参数。

内置模型和自定义模型属性展示

右边区域框内的"property"即为属性设置栏,选中Model节点即可对其属性进行设置,如图5-26所示。

图 5-26 Model 属性设置



节点属性说明如下:

- Name: Model名,在New Model时决定,不能修改。
- Model Path: Model存放在Mind Studio上的路径,在New Model时决定,不能修改。

- **DVPP Parameter Path**: DVPP调优参数文件存放在Mind Studio上的路径,由Dvpp tuning生成。
- **Decryption:** 若Model是加密的,打开Decryption开关并上传解密秘钥,详细信息请参见**5.2.3** 自定义模型加解密。

Caffe 模型属性展示

Caffe 模型属性展示如图5-27所示。

图 5-27 caffe 模型属性



节点的属性主要分为:

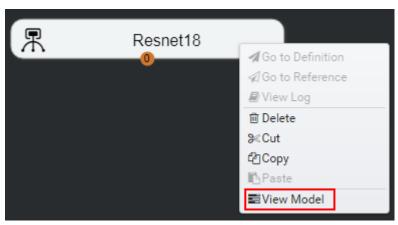
- Name: Model名,在新增Caffe Model时输入,不能修改。
- **Model File**:模型文件存放在Mind Studio上的路径,在New Model时决定,不能修改。
- Weight File: 权重文件存放在Mind Studio上的路径,在New Model时决定,不能修改。

5.2.6 模型网络结构查看

内置模型和自定义模型网络结构查看

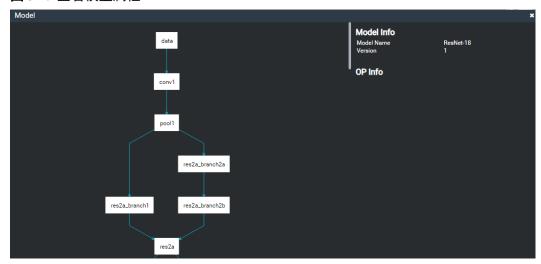
将离线模型组件拖入画布,在画布右键单击模型,选择"View Model",如图5-28所示。

图 5-28 查看模型



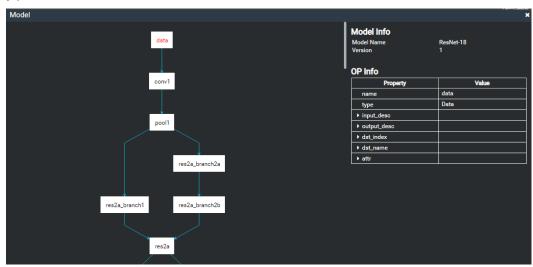
弹出离线模型网络拓扑结构对话框,如图5-29所示。右侧上方展示模型的基本信息:

图 5-29 查看模型属性



单击模型的某一层,可以展示该层的详细属性,如图5-30所示。

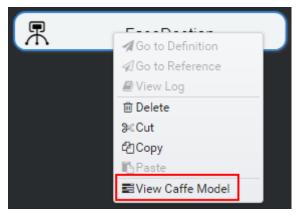
图 5-30 查看详细属性信息



Caffe 模型网络结构查看

将Caffe模型组件拖入画布,在画布右键单击"View Caffe Model",如图5-31所示。

图 5-31 查看 caffe 模型



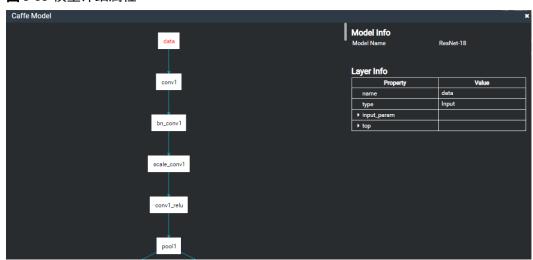
弹出离线模型网络拓扑结构对话框,如图5-32所示。右侧上方展示模型的基本信息:

图 5-32 展示模型基本信息



单击模型的某一层,可以展示该层的详细属性,如图5-33所示。

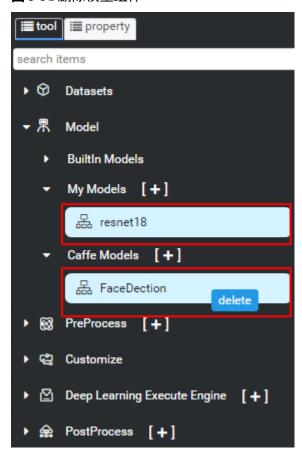
图 5-33 模型详细属性



5.2.7 模型组件删除

自定义模型组件和Caffe模型组件都可以删除,右键单击需要删除的模型组件,单击"delete"即可删除该组件,如果画布上已经使用了该组件,需要先删除画布上的组件,如图5-34所示。

图 5-34 删除模型组件



□说明

My Model或者Caffe Model组件删除的同时,也会删除model-zoo下的离线模型或Caffe模型文件。

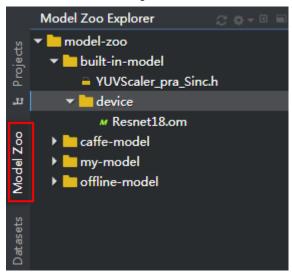
5.3 代码编辑窗口的模型管理

5.3.1 模型展示

单击左侧 "Model Zoo" 页签会出现 "Model Zoo Explorer" 对话框,该Explorer展示 models的目录信息,分为3个子目录,如图5-35所示。

- built-in-model,即内置模型。
- my-model,即自定义模型。
- caffe-model,即Caffe模型。

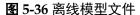


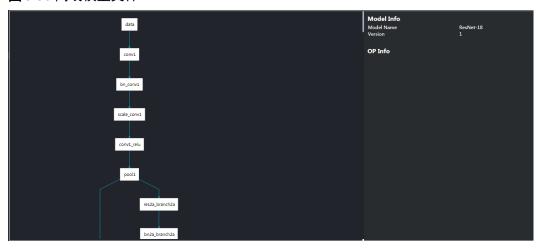


单击 能看到目录具体内容,例如上图中单击built-in-model旁边的图标,展示内置模型。

模型网络结构查看

在Model Zoo Explorer中双击某个离线模型文件(仅支持离线模型,请勿双击caffe模型的文件),如上图中的Resnet18.om文件,在右侧编辑器中打开离线模型,展示详细信息,如图5-36所示。





左侧展示模型的网络拓扑结构图,右侧上方展示模型名称和版本号。单击模型的某一层,可以展示该层的详细属性列表,如图5-37所示。



图 5-37 模型详细属性

5.3.2 新增自定义模型组件

代码编辑窗口下新增自定义模型组件的方法请参考 "Mind Engine的模型管理 > 新增自定义模型组件"中的**通过模型转换方式生成**。

5.3.3 Copy Path 功能

选择一个Model文件,单击鼠标右键,会出现Copy Path,如图5-38所示,单击Copy Path,会把后台服务器中文件的路径复制到剪切板,以供用户使用该Model。

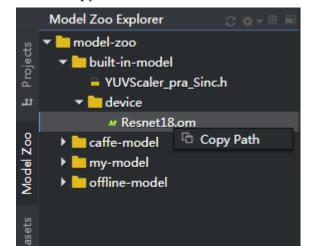


图 5-38 Copy Path

5.3.4 Refresh 功能

新增我的模型或者在菜单界面转换模型等操作,会在Model Zoo中生成新的文件夹(文件)。选择一个文件夹,如图5-39,单击Refresh图标,会刷新该路径下一层的信息(如果展开多层,也只刷新一层,因为有可能目录中文件过多,造成性能问题),刷新后如图5-40。

图 5-39 Refresh 前

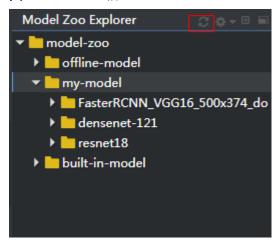
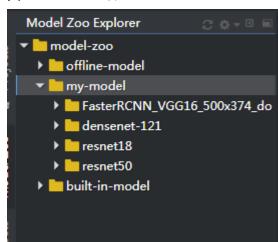


图 5-40 Refresh 后



□说明

基于性能原因考虑,暂不支持通过展开折叠文件夹的方式刷新目录,如果要刷新目录,只能通过 单击■实现。

5.3.5 离线模型导入

对于用户本地已经转化好的离线模型文件,可以导入到工程目录供工程开发使用,操作如下:

步骤1 离线模型导入。

选中要导入模型的工程,通过右侧tool栏"Model > My Model",单击"+",弹出"New Model"窗口,界面参数解释如表5-3所示。导入后的界面如图5-41所示。

图 5-41 添加 OfflineModel

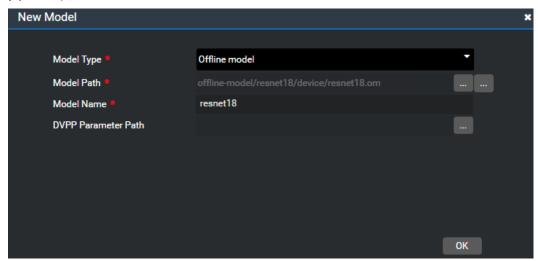


表 5-3 离线模型导入界面参数

界面参数	取值
"Model Type"	选择"OfflineModel"。
"Model Path"	"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"。

步骤2 模型文件应用一Engine编排。

离线模型导入后,在右侧"My Models"窗口中可以看到导入的模型,如图5-42所示。可以将该模型拖拽到画布中进行Engine编排。

图 5-42 My Models 窗口



步骤3 模型文件应用一自定义工程。

导入工程中的自定义模型,在代码开发中可以直接引用模型文件所在路径。

----结束

5.4 参考

5.4.1 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。
# 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
# 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
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
```

5.4.2 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5-4。

表 5-4	二进制	l 文件:	ねポ
4X J-4	V ' m'	ידועו	ロル

Offset	type	value	description
0000	32bit int	magic	magic number
0004	32bit int	50	input num
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 5-5 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 5-6 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 5-7 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 5-8 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 5-9 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 5-10 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 5-11 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 5-12 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 5-13 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 5-14 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 5-15 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 5-16 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 5-17 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。

表 5-18 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 5-19 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 5-20 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 5-21 start_ratio、end_ratio、step_ratio 参数说明

名称	参数说明
类型	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 5-22 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 5-23 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	 只支持Conv、FC、DepthWiseConv算子; 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