

密级状态：绝密() 秘密() 内部() 公开(√)

RKNN-Toolkit 快速上手指南

(技术部，图形显示平台中心)

文件状态： [] 正在修改 [√] 正式发布	当前版本：	V1.0.0
	作 者：	饶洪
	完成日期：	2019-05-08
	审 核：	卓鸿添
	完成日期：	2019-05-08

福州瑞芯微电子股份有限公司

Fuzhou Rockchips Semiconductor Co., Ltd

(版本所有, 翻版必究)

更新记录

版本	修改人	修改日期	修改说明	核定人
V0.9.9	饶洪	2019-03-25	初始版本	卓鸿添
V1.0.0	饶洪	2019-05-08	同步 RKNN-Toolkit-V1.0.0 修改内容	卓鸿添

目 录

1	主要功能说明.....	1
2	系统依赖说明.....	2
3	快速上手.....	3
3.1	环境准备	3
3.2	安装 RKNN-TOOLKIT（以 PYTHON3.5 为例）	3
3.3	运行安装包中附带的示例.....	4
3.3.1	在 PC 上仿真运行示例.....	4
3.3.2	在 RK1808 上运行示例.....	6

1 主要功能说明

RKNN-Toolkit 是为用户提供在 PC、RK3399Pro、RK1808 或 RK3399Pro Linux 开发板上进行模型转换、推理和性能评估的开发套件，用户通过提供的 python 接口可以便捷地完成以下功能：

1) 模型转换：支持 Caffe、TensorFlow、TensorFlow Lite、ONNX、Darknet 模型转成 RKNN 模型，支持 RKNN 模型导入导出，后续能够在硬件平台上加载使用。

2) 模型推理：能够在 PC 上模拟运行模型并获取推理结果；也可以在指定硬件平台 RK3399Pro（或 RK3399Pro Linux 开发板）、RK1808 上运行模型并获取推理结果。

3) 性能评估：能够在 PC 上模拟运行并获取模型总耗时及每一层的耗时信息；也可以通过联机调试的方式在指定硬件平台 RK3399Pro、RK1808 上运行模型，或者直接在 RK3399Pro Linux 开发板上运行，以获取模型在硬件上完整运行一次所需的总时间和每一层的耗时情况。

4) 获取模型运行时的内存使用情况：通过联机调试的方式获取模型在指定硬件平台 RK3399Pro、RK1808 或 RK3399Pro Linux 开发板上运行时的内存使用情况。

5) 量化功能：支持将浮点模型转成量化模型，目前支持的量化方法有非对称量化（asymmetric_quantized-u8），动态定点量化（dynamic_fixed_point-8 和 dynamic_fixed_point-16）。

2 系统依赖说明

本开发套件支持运行于 Ubuntu 操作系统。需要满足以下运行环境要求：

表 1 运行环境

操作系统版本	Ubuntu16.04（x64）以上
Python 版本	3.5/3.6
Python 库依赖	'numpy >= 1.16.1' 'scipy >= 1.1.0' 'Pillow >= 3.1.2' 'h5py >= 2.7.1' 'lmbd >= 0.92' 'networkx == 1.11' 'flatbuffers == 1.9', 'protobuf >= 3.5.2' 'onnx >= 1.3.0' 'flask >= 1.0.2' 'tensorflow >= 1.11.0' 'dill==0.2.8.2' 'opencv-python>=3.4.3.18' 'ruamel.yaml==0.15.82'

3 快速上手

本章节以 RK1808、Python3.5 为例说明如何快速上手使用 RKNN-Toolkit。

3.1 环境准备

- 一台安装有 ubuntu16.04 操作系统的 x86_64 位计算机。
- 一个搭载 RK1808 的硬件平台。
- 将 RK1808 设备通过 USB 连接到 PC 上，使用 adb devices 命令查看，结果如下：

```
rk@rk:~$ adb devices
List of devices attached
0123456789ABCDEF    device
```

3.2 安装 RKNN-Toolkit（以 Python3.5 为例）

1. 安装 Python3.5

```
sudo apt-get install python3.5
```

2. 安装 pip3

```
sudo apt-get install python3-pip
```

3. 获取 RKNN-Toolkit 安装包，然后执行以下步骤：

- a) 进入 package 目录：

```
cd package/
```

- b) 安装 Python 依赖

```
sudo pip3 install -r requirements-cpu.txt
```

- c) 安装 RKNN-Toolkit

```
sudo pip3 install rknn_toolkit-1.0.0-cp35-cp35m-linux_x86_64.whl
```

d) 检查 RKNN-Toolkit 是否安装成功

```
rk@rk:~/rknn-toolkit-v1.0.0/package$ python3
>>> from rknn.api import RKNN
>>>
```

如果导入 RKNN 模块没有失败，说明安装成功。

3.3 运行安装包中附带的示例

3.3.1 在 PC 上仿真运行示例

RKNN-Toolkit 自带了一个 RK1808 的模拟器，可以用来仿真模型在 RK1808 上运行时的行为。

这里以 mobilenet_v1 为例。示例中的 mobilenet_v1 是一个 Tensorflow Lite 模型，用于图片分类，它是在模拟器上运行的。

运行该示例的步骤如下：

1. 进入 example/mobilenet_v1 目录

```
rk@rk:~/rknn-toolkit-v1.0.0/package$ cd ../example/mobilenet_v1
rk@rk:~/rknn-toolkit-v1.0.0/example/mobilenet_v1$
```

2. 执行 test.py 脚本

```
rk@rk:~/rknn-toolkit-v1.0.0/example/mobilenet_v1$ python3 test.py
```

3. 脚本执行完后得到如下结果：

```
--> config model
done
--> Loading model
done
--> Building model
done
--> Export RKNN model
done
--> Init runtime environment
done
```

```
--> Running model
mobilenet_v1
-----TOP 5-----
[156]: 0.8837890625
[155]: 0.0677490234375
[188 205]: 0.00867462158203125
[188 205]: 0.00867462158203125
[263]: 0.0057525634765625

done
--> Begin evaluate model performance
=====
                                Performance
=====
Layer ID      Name                                     Time(us)
0             tensor.transpose_3                      72
45            convolution.relu.pooling.layer2_2         363
60            convolution.relu.pooling.layer2_2         200
46            convolution.relu.pooling.layer2_2         185
61            convolution.relu.pooling.layer2_2         242
47            convolution.relu.pooling.layer2_2          98
62            convolution.relu.pooling.layer2_2         149
48            convolution.relu.pooling.layer2_2         152
63            convolution.relu.pooling.layer2_2         120
49            convolution.relu.pooling.layer2_2         116
64            convolution.relu.pooling.layer2_2         101
50            convolution.relu.pooling.layer2_2         185
65            convolution.relu.pooling.layer2_2         101
51            convolution.relu.pooling.layer2_2         111
66            convolution.relu.pooling.layer2_2         109
52            convolution.relu.pooling.layer2_2         213
67            convolution.relu.pooling.layer2_2         109
53            convolution.relu.pooling.layer2_2         213
68            convolution.relu.pooling.layer2_2         109
54            convolution.relu.pooling.layer2_2         213
69            convolution.relu.pooling.layer2_2         109
55            convolution.relu.pooling.layer2_2         213
70            convolution.relu.pooling.layer2_2         109
56            convolution.relu.pooling.layer2_2         213
71            convolution.relu.pooling.layer2_2         109
57            convolution.relu.pooling.layer2_2         174
72            convolution.relu.pooling.layer2_2         219
58            convolution.relu.pooling.layer2_2         353
59            fullyconnected.relu.layer_3             110
30            tensor.transpose_3                        5
Total Time(us): 4775
FPS(800MHz): 209.42
=====

done
```


这个例子涉及到的主要操作有：创建 RKNN 对象；模型配置；加载 TensorFlow Lite 模型；构建 RKNN 模型；导出 RKNN 模型；加载图片并推理，得到 TOP5 结果；评估模型性能；释放 RKNN 对象。

example 目录中的 mobilenet_v2 和 mobilenet-ssd 示例的执行方式与 mobilenet_v1 相同，只是 mobilenet-ssd 示例中要执行的脚本是 ssd.py，且该示例执行完后会输出一张 out.jpg 图像，它会把检测到的对象在图像中用方框框出来。

3.3.2 在 RK1808 上运行示例

这里以 mobilenet_v1 为例。工具包中带的 mobilenet_v1 示例是在 PC 模拟器上运行的，如果要在 RK1808 设备上运行这个示例，可以参考以下步骤：

1. 进入 example/mobilenet_v1 目录

```
rk@rk:~/rknn-toolkit-v1.0.0/example/mobilenet_v1$
```

2. 修改 test.py 脚本里的初始化环境变量时带的参数

```
rk@rk:~/rknn-toolkit-v1.0.0/example/mobilenet_v1$ vim test.py
# 找到脚本里初始化环境变量的方法 init_runtime，如下
ret = rknn.init_runtime()
# 修改该方法的参数
ret = rknn.init_runtime(target='rk1808', device_id='0123456789ABCDEF')
# 保存修改并退出
```

3. 执行 test.py 脚本，得到如下结果：

```
rk@rk:~/rknn-toolkit-v1.0.0/example/mobilenet_v1$ python test.py
--> config model
done
--> Loading model
done
--> Building model
done
--> Export RKNN model
done
--> Init runtime environment
done
--> Running model
```

```
mobilenet_v1
-----TOP 5-----
[156]: 0.8837890625
[155]: 0.0677490234375
[188 205]: 0.00867462158203125
[188 205]: 0.00867462158203125
[263]: 0.0057525634765625

done
--> Begin evaluate model performance
=====
                                Performance
=====
Total Time(us): 5840
FPS: 171.23
=====

done
```

有关 RKNN-Toolkit 更详细的用法和接口说明，请参考《RKNN-Toolkit 使用指南_V1.0.0.pdf》手册。