

# Paddle Lite 模型 Android NNAPI 部署指导手册

文档版本  
发布日期


V1.1  
2023-05-12

## 版权所有 © 紫光展锐（上海）科技有限公司。保留一切权利。

本文件所含数据和信息都属于紫光展锐（上海）科技有限公司（以下简称紫光展锐）所有的机密信息，紫光展锐保留所有相关权利。本文件仅为信息参考之目的提供，不包含任何明示或默示的知识产权许可，也不表示有任何明示或默示的保证，包括但不限于满足任何特殊目的、不侵权或性能。当您接受这份文件时，即表示您同意本文件中内容和信息属于紫光展锐机密信息，且同意在未获得紫光展锐书面同意前，不使用或复制本文件的整体或部分，也不向任何其他方披露本文件内容。紫光展锐有权在未经事先通知的情况下，在任何时候对本文件做任何修改。紫光展锐对本文件所含数据和信息不做任何保证，在任何情况下，紫光展锐均不负任何与本文件相关的直接或间接的、任何伤害或损失。

请参照交付物中说明文档对紫光展锐交付物进行使用，任何人 对紫光展锐交付物的修改、定制化或违反说明文档的指引对紫光展锐交付物进行使用造成的任何损失由其自行承担。紫光展锐交付物中的性能指标、测试结果和参数等，均为在紫光展锐内部研发和测试系统中获得的，仅供参考，若任何人需要对交付物进行商用或量产，需要结合自身的软硬件测试环境进行全面的测试和调试。

## 商标声明

**紫光展锐**、**UNISOC**、、展讯、Spreadtrum、SPRD、锐迪科、RDA 及其他紫光展锐的商标均为紫光展锐（上海）科技有限公司及/或其子公司、关联公司所有。

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。

## 免责声明

本文档可能包含第三方内容，包括但不限于第三方信息、软件、组件、数据等。紫光展锐不控制且不对第三方内容承担任何责任，包括但不限于准确性、兼容性、可靠性、可用性、合法性、适当性、性能、不侵权、更新状态等，除非本文档另有明确说明。在本文档中提及或引用任何第三方内容不代表紫光展锐对第三方内容的认可、承诺或保证。

用户有义务结合自身情况，检查上述第三方内容的可用性。若需要第三方许可，应通过合法途径获取第三方许可，除非本文档另有明确说明。

# 紫光展锐（上海）科技有限公司



# 前言

## 概述

本文档主要介绍了在紫光展锐芯片上通过 Paddle Lite 将 Paddle 模型部署到 Android 设备上，并通过调用 Android NNAPI 实现硬件加速推理运算，内容涵盖环境准备，demo 和模型准备、相应设置及运行结果示例等。

## 读者对象




本文档主要适用于需要部署 Paddle 模型的开发人员。

## 缩略语

缩略语	英文全称	中文解释
NNAPI	Neural Networks API	神经网络 API

## 符号约定

在本文中可能出现下列符号，每种符号的说明如下。

符号	说明
 <b>说明</b>	用于突出重要或关键信息、补充信息和小窍门等。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害。
 <b>注意</b>	用于突出容易出错的操作。 “注意”不是安全警示信息，不涉及人身、设备及环境伤害。
 <b>警告</b>	用于可能无法恢复的失误操作。 “警告”不是危险警示信息，不涉及人身及环境伤害。

## 变更信息

文档版本	发布日期	修改说明
V1.1	2023-05-12	<ul style="list-style-type: none"><li>调整文档架构，优化文档内容。</li><li>将文档名更新为《Paddle Lite 模型 Android NNAPI 部署指导手册》。</li></ul>
V1.0	2022-10-19	第一次正式发布

## 关键字

Paddle lite、Android NNAPI

---

# 目 录

---

1 环境准备.....	1
2 Android NNAPI 部署 .....	2
2.1 demo 准备 .....	2
2.2 模型准备 .....	2
2.3 源码中使用 NNAPI.....	2
2.4 shell 脚本添加 NNAPI 设置.....	3
2.5 运行结果 .....	3
3 参考文档.....	6

# 1 环境准备

---

- 电脑或者服务器，确保已安装 ADB 调试工具，且 ADB 命令可用。
- 测试设备（Android 13）。
- USB 数据线，用于连接测试设备和电脑或者服务器。

# 2 Android NNAPI 部署

Paddle Lite 封装完善，可以通过配置相应的 config 参数来实现 Android NNAPI（Neural Networks API，神经网络 API）部署。

Paddle Lite 框架提供了 Android 平台的官方 Release 预测库下载，可根据相应平台和运行需求下载相适配的 Paddle Lite 预编译库版本。

本文以 image\_classification\_demo 为例进行说明。

## 2.1 demo 准备

步骤 1 下载 demo 源码包：<https://paddlite-demo.bj.bcebos.com/devices/generic/PaddleLite-generic-demo.tar.gz>。

步骤 2 解压 PaddleLite-generic-demo.tar.gz 后得到 PaddleLite-generic-demo 文件夹。

### 说明

文件夹包含了所有 Paddle Lite 的依赖库 so 文件以及相关的运行代码。

步骤 3 进入其中的/image\_classification\_demo/shell/文件夹，需要更改配置的文件如下：

- run\_with\_adb.sh，示例程序 ADB 运行脚本，具体修改详见 [2.4 shell 脚本添加 NNAPI 设置](#)。
- image\_classification\_demo.cc，示例程序源码的入参，具体修改详见 [2.3 源码中使用 NNAPI](#)。

----结束

## 2.2 模型准备

步骤 1 下载 Paddle Lite 模型：

[https://paddlite-demo.bj.bcebos.com/models/mobilenet\\_v1\\_int8\\_224\\_per\\_layer.tar.gz](https://paddlite-demo.bj.bcebos.com/models/mobilenet_v1_int8_224_per_layer.tar.gz)

步骤 2 将 mobilenet\_v1\_int8\_224\_per\_layer.tar.gz 解压后得到 mobilenet\_v1\_int8\_224\_per\_layer 文件夹。

步骤 3 将 mobilenet\_v1\_int8\_224\_per\_layer 文件夹移动至 PaddleLite-generic-demo/image\_classification\_demo/assets/models 文件夹下

----结束

## 2.3 源码中使用 NNAPI

Paddle Lite 在入参时指定 nnadapter\_device\_names 为 android\_nnapi 即可调用 NNAPI 进行推理。

以 `image_classification_demo.cc` 为例，无需改动源码，请将入参 `nnadapter_device_names` 设置为 `android_nnapi`，通过函数 `set_nnadapter_device_names()` 赋值给 `CxxConfig` 和 `mobile_config`，示例如下：

```
int main(int argc, char **argv) {
    if (argc < 10) {
        ...
    }
    ...
    std::vector<std::string> nnadapter_device_names =
        split_string<std::string>(argv[5], ',');
    ...
    // Run inference by using full api with CxxConfig
    paddle::lite_api::CxxConfig cxx_config;
    cxx_config.set_nnadapter_device_names(nnadapter_device_names);
    // Run inference by using light api with MobileConfig
    paddle::lite_api::MobileConfig mobile_config;
    mobile_config.set_nnadapter_device_names(nnadapter_device_names);
}
```

### 注意

同时添加多个设备会影响 NNAPI 的推理性能，比如不建议使用 `nnadapter_device_names='cpu','android_nnapi'`，而建议使用 `nnadapter_device_names='android_nnapi'` 来进行部署。

## 2.4 shell 脚本添加 NNAPI 设置

在 `run_with_adb.sh` 文件中将原脚本的运行后端 “`cpu`” 修改为 “`android_nnapi`”：

```
#NNADAPTER_DEVICE_NAME="cpu"
NNADAPTER_DEVICE_NAME="android_nnapi"
```

## 2.5 运行结果

在 Linux 终端下运行如下命令：

```
cd PaddleLite-generic-demo/image_classification_demo/shell
./run_with_adb.sh
```

以 UMS9620 平台为例，NNAPI 与 CPU 运行结果如下。

- NNAPI 运行结果

```
...
iter 0 cost: 9.643000 ms
```



```

iter 1 cost: 11.917000 ms
iter 2 cost: 13.156000 ms
iter 3 cost: 13.061000 ms
iter 4 cost: 21.657000 ms
warmup: 1 repeat: 5, average: 13.886800 ms, max: 21.657000 ms, min: 9.643000 ms

results: 3
Top0 Egyptian cat - 0.450432
Top1 tabby, tabby cat - 0.450432
Top2 tiger cat - 0.087746
Preprocess time: 0.685000 ms
Prediction time: 13.886800 ms
Postprocess time: 0.897000 ms
    
```

上述结果中，NNAPI 推理时间为 13.886800 ms。

- CPU 运行结果

```

...
iter 0 cost: 39.889000 ms
iter 1 cost: 40.167999 ms
iter 2 cost: 40.193001 ms
iter 3 cost: 40.099998 ms
iter 4 cost: 40.428001 ms
warmup: 1 repeat: 5, average: 40.155600 ms, max: 40.428001 ms, min: 39.889000 ms

results: 3
Top0 Egyptian cat - 0.512545
Top1 tabby, tabby cat - 0.402567
Top2 tiger cat - 0.067904
Preprocess time: 0.673000 ms
Prediction time: 40.155600 ms
Postprocess time: 0.107000 ms
    
```

上述结果中，CPU 推理时间为 40.155600 ms。

主要运行结果说明见下表。

运行结果	说明
iter 0 ~ 5 cost	每次的推理时间，单位 ms。
warmup:1	第一次运行时间，不算在推理时间内。

运行结果	说明
repeat:5	在 warmup 之后，重复推理运行 5 次（5 为默认值）。
average	5 次的平均推理时间，单位 ms。
max	5 次中最长推理时间，单位 ms。
min	5 次中最短推理时间，单位 ms。
results 3	代表推理运行会输出概率最大的 top3 的结果以及对应的 score，即 log 中的 Top0 Egyptian cat、Top1 tabby, tabby cat 和 Top2 tiger cat 及对应的 score。
Preprocess time	预处理时间，单位 ms。
Prediction time	平均推理时间，与 average 相同，单位 ms。
Postprocess time	后处理时间，单位 ms。

## 📖 说明

- “tabby, tabby cat” 表示 demo 选取的输入图片为 tabby\_cat.jpg。
- 不同运行环境下的运行结果可能有细微出入，若需要替换模型或更改图片等请参考 [https://www.paddlepaddle.org.cn/lite/v2.11/demo\\_guides/android\\_nnapi.html](https://www.paddlepaddle.org.cn/lite/v2.11/demo_guides/android_nnapi.html)。

# 3 参考文档

---

Android NNAPI 部署示例: [https://www.paddlepaddle.org.cn/lite/v2.11/demo\\_guides/android\\_nnapi.html](https://www.paddlepaddle.org.cn/lite/v2.11/demo_guides/android_nnapi.html)