一个面向RISC-V的深度学习推理框架的设计与实现

侯朋朋 于佳耕 苗玉霞 郤阳 武延军 赵琛

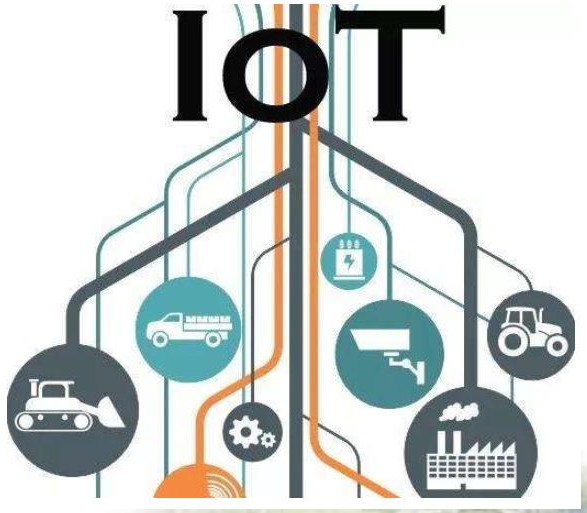
2019年11月

*本文成果首发于 2019 BenchCouncil International Artificial Intelligence System Challenges

背景

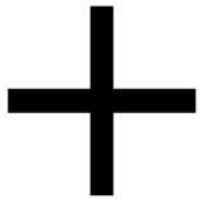
- **RISC-V指令集发展迅速**
  - 基于精简指令集原则的开源指令集架构

- **RISC-V指令集很适合IoT场景**
  - 基本指令集+扩展指令集
  - IoT场景碎片化



简介

- 当前主流的推理系统
  - 服务器：TensorFlow、 MXNet 、Caffe
  - 智能终端：TensorFlow Lite、NCNN、MNN



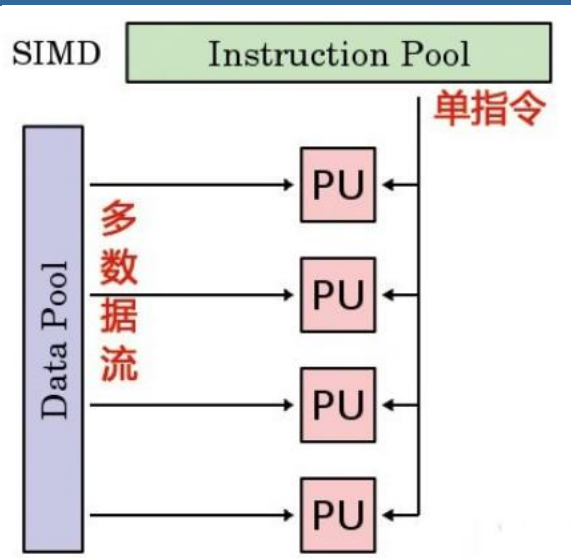
intel

NVIDIA

ARM

简介

- 面向RISC-V + IoT的推理系统还很少
  - 体系架构紧绑定的库不可用
    - SIMD特性
  - IoT硬件资源紧张
    - 内存和存储远小于服务器和智能终端
    - IoT设备成本远低于智能终端



安防监控摄像头价格区间

价格区间	90~150	150~775	大于775
用户比例	34%	37%	29%

简介

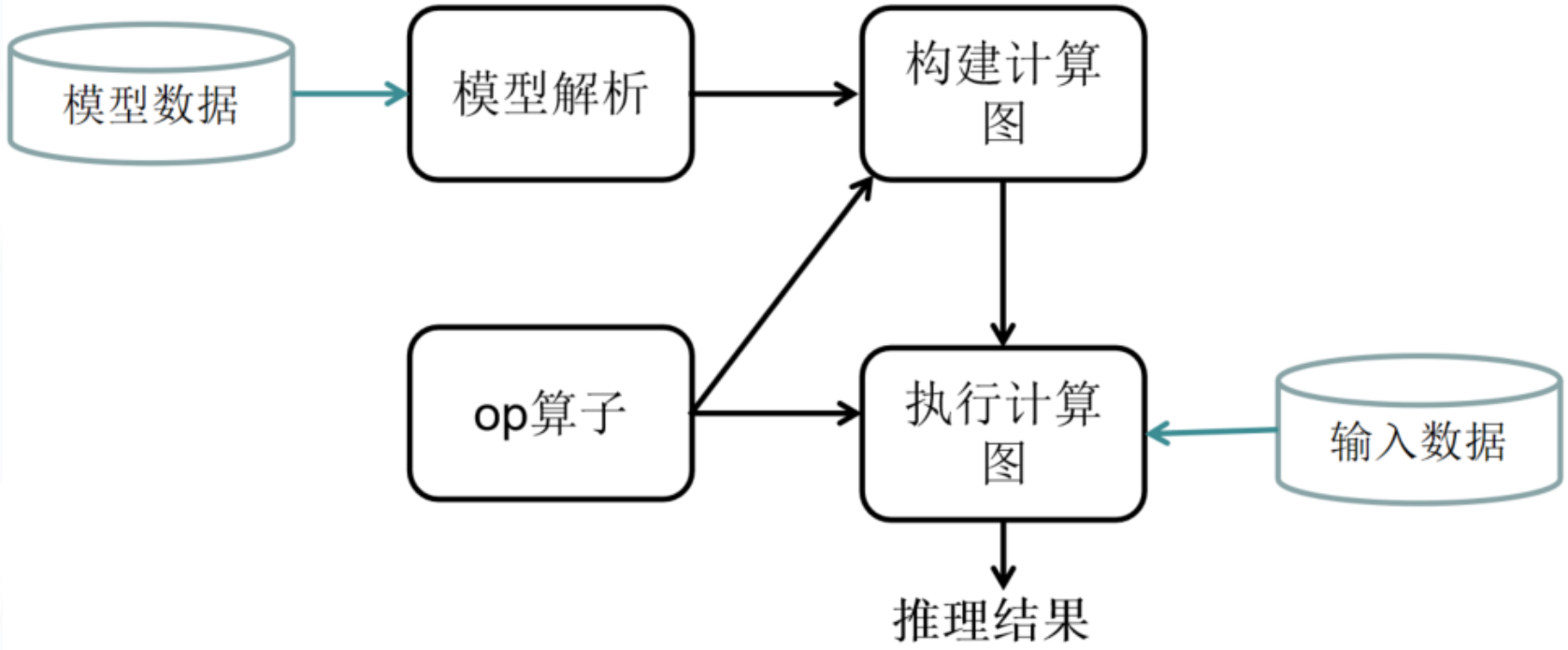
- **RVTensor：RISC-V Tensor**
  - 面向RISC-V + IoT的深度学习推理框架
  - 依赖第三方库少
    - 仅依赖H5模型解析的libhd5.so
  - 内存等资源需求少
  - 基于思沃r版 (SERVE.r) 实现

全系统平台配置

- Rocket单核/双核@50-100MHz
- UART、GbE、SDIO、USB、HDMI外设
- Linux v4.19 + Debian社区生态
- FPGA定制加速
- 低成本+低功耗板卡

总体架构

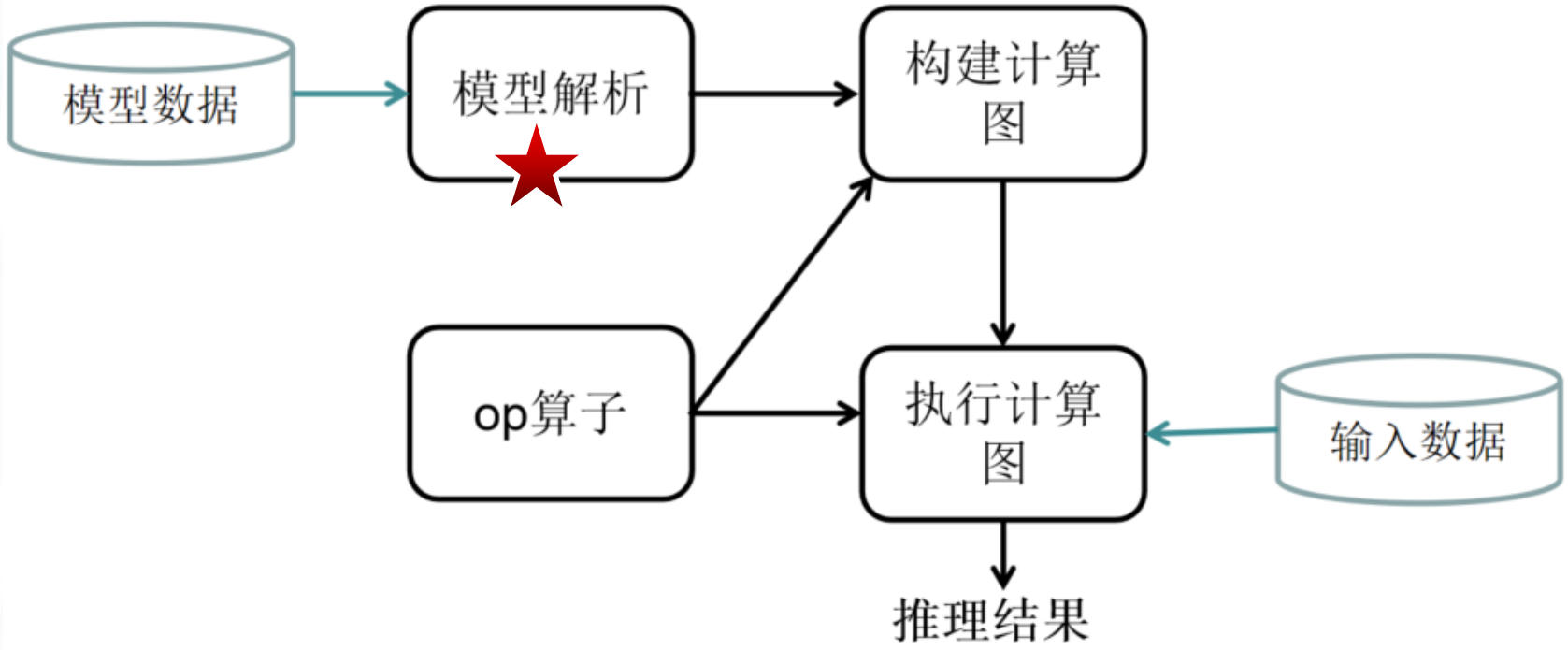
RVTensor架构



- 四个部分组成：模型解析、OP算子、构建计算图、执行计算图

总体架构

RVTensor架构

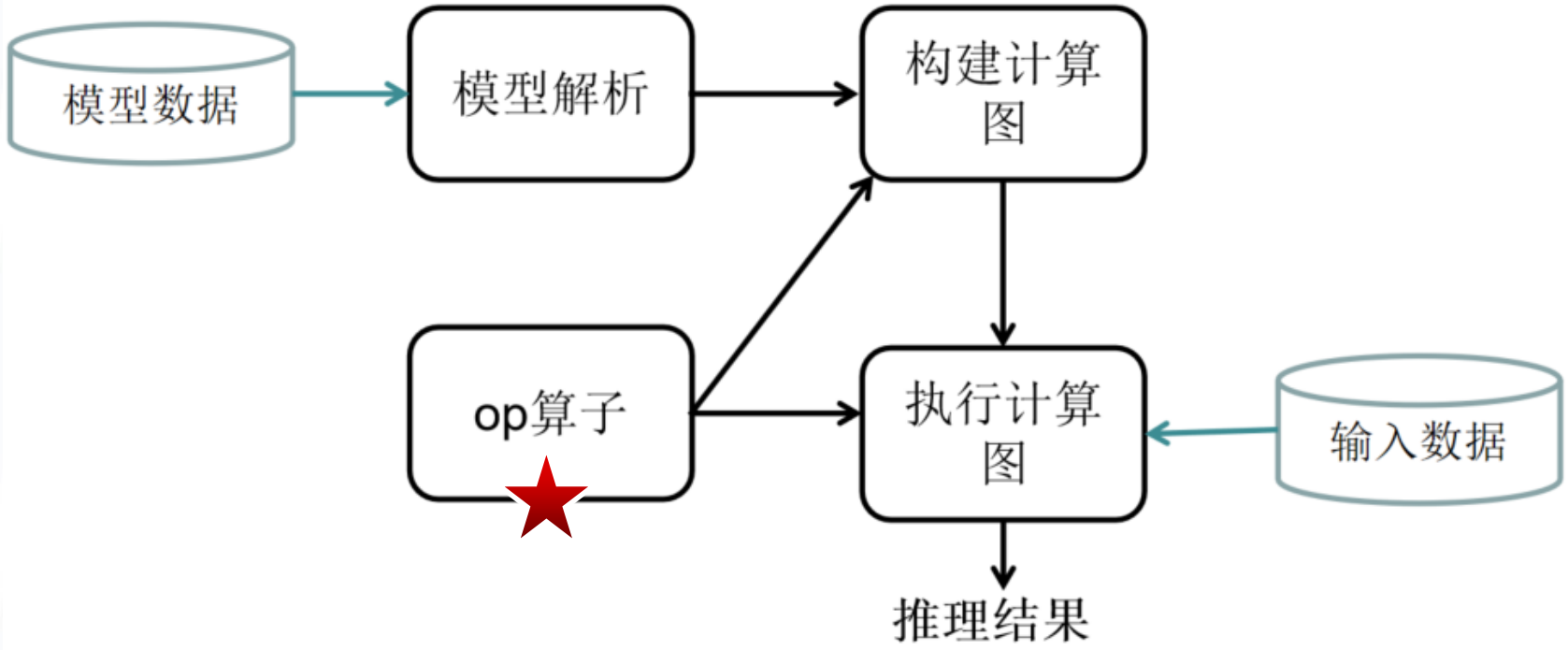


模型解析

- 主要对模型文件如.pb进行解析，读取算子操作、权值数据等信息

总体架构

RVTensor架构

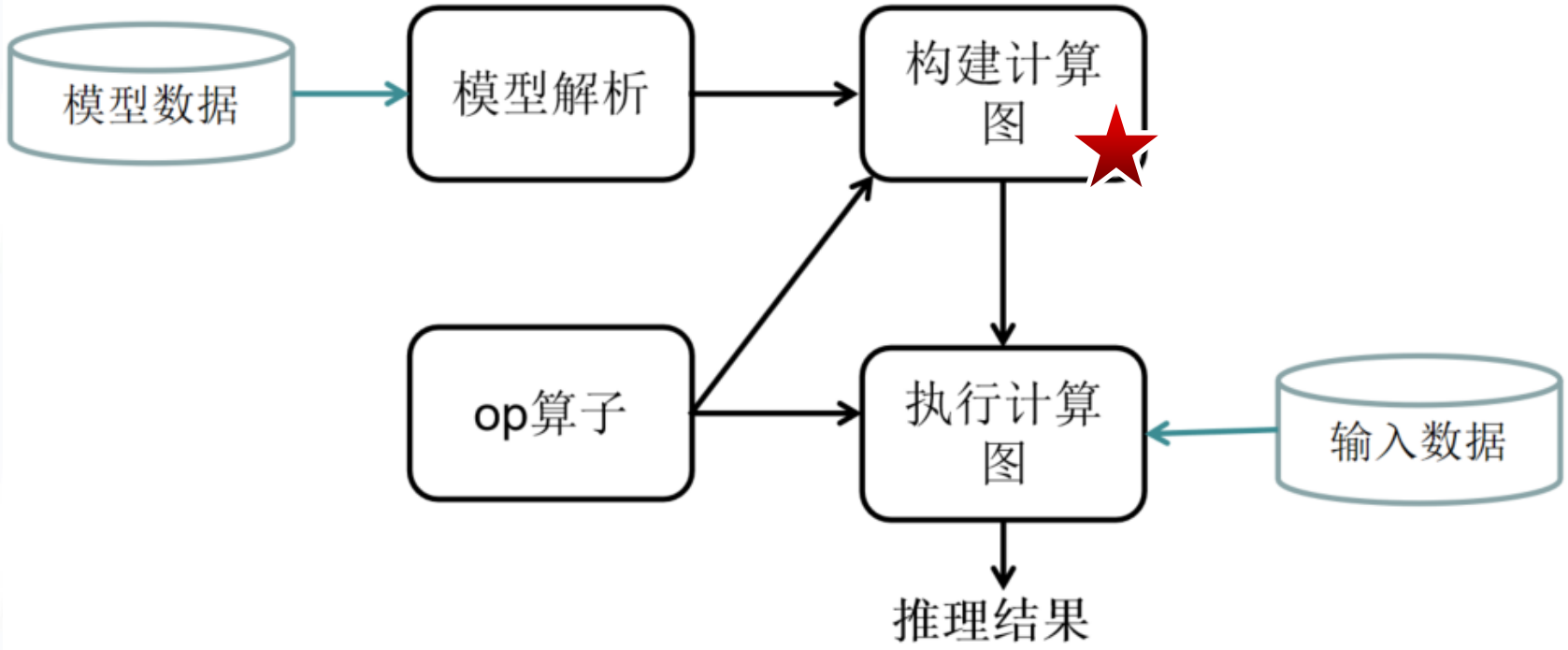


OP算子

- 包括conv、add、active、pooling等算子

总体架构

RVTensor架构

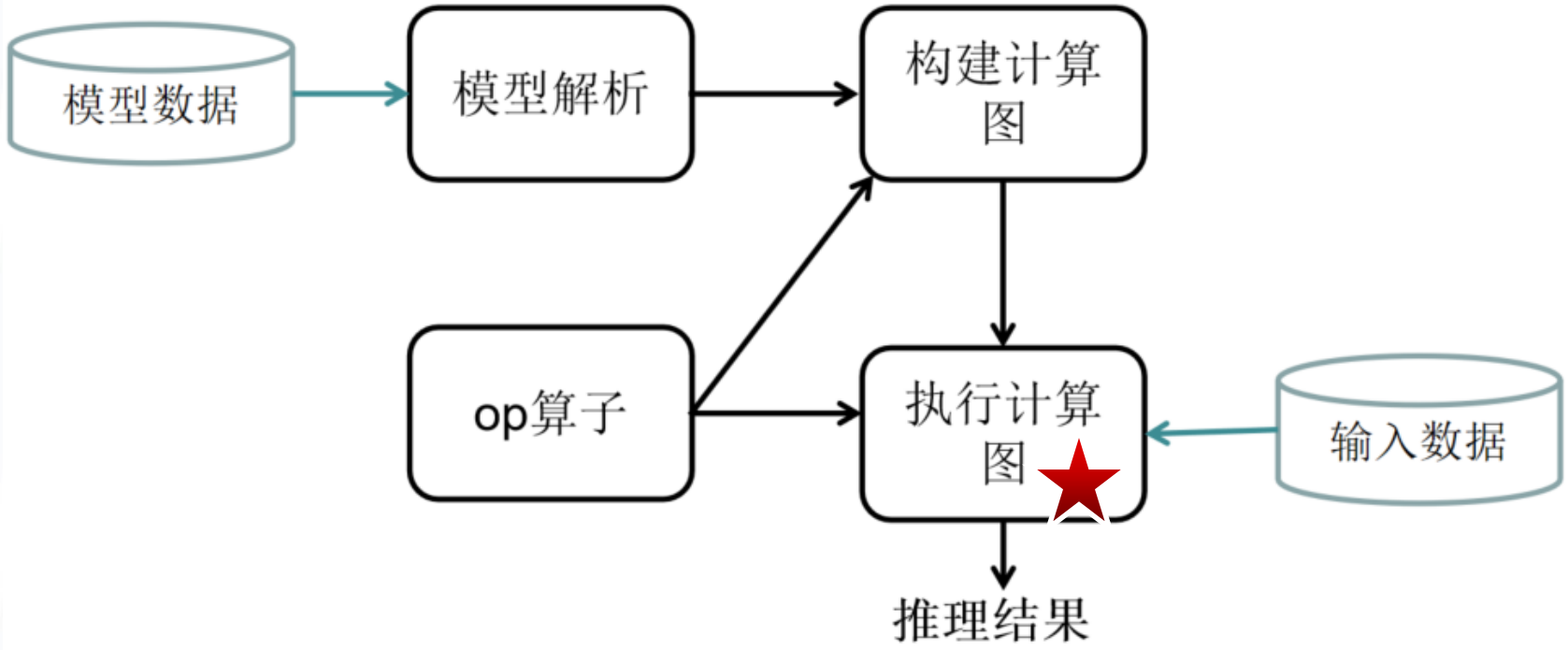


构建计算图

- 基于模型解析和op算子模块构建出计算图

总体架构

RVTensor架构

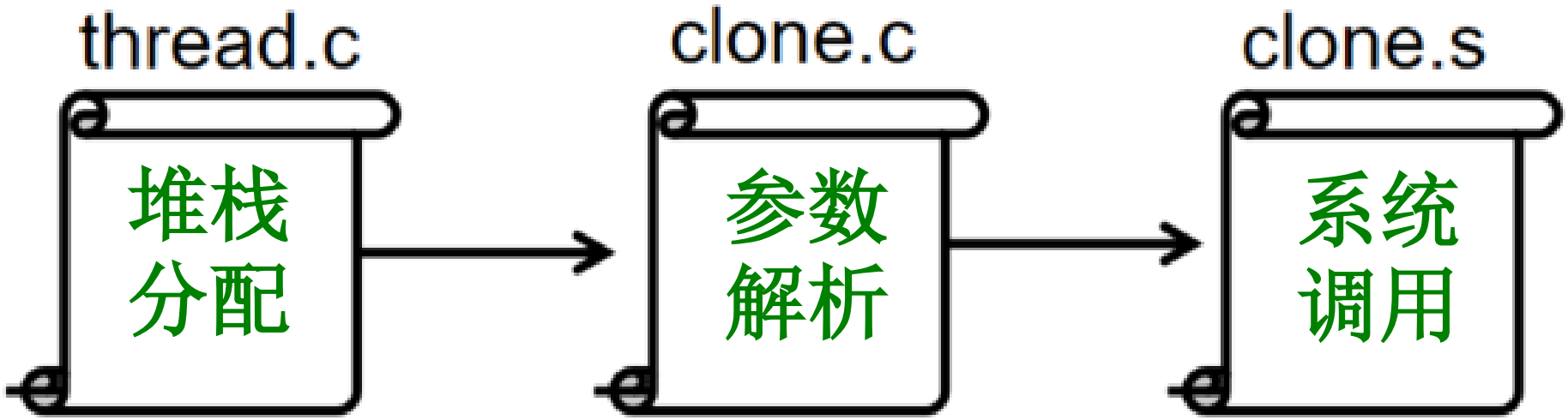


执行计算图

- 该模块基于输入数据和计算图进行计算并得到推理结果

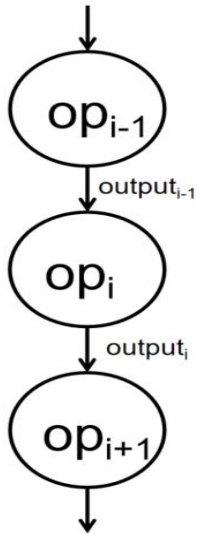
典型优化技术

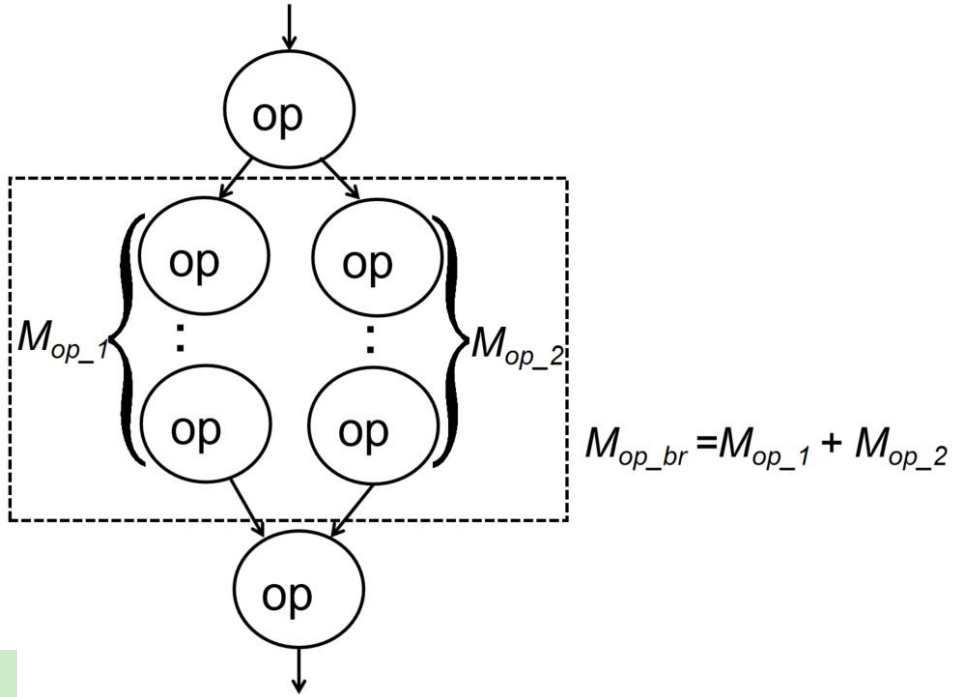
- 针对对第三方库的依赖优化
  - 典型的工作是优化多线程库 **Pthread**
    - 功能很全面
    - 但是用到的**API**很少



典型优化技术

- 内存优化
  - 内存复用：所有op运行时复用同一块内存
    - 最大的op占用内存量
    - 分支部分当做原子op

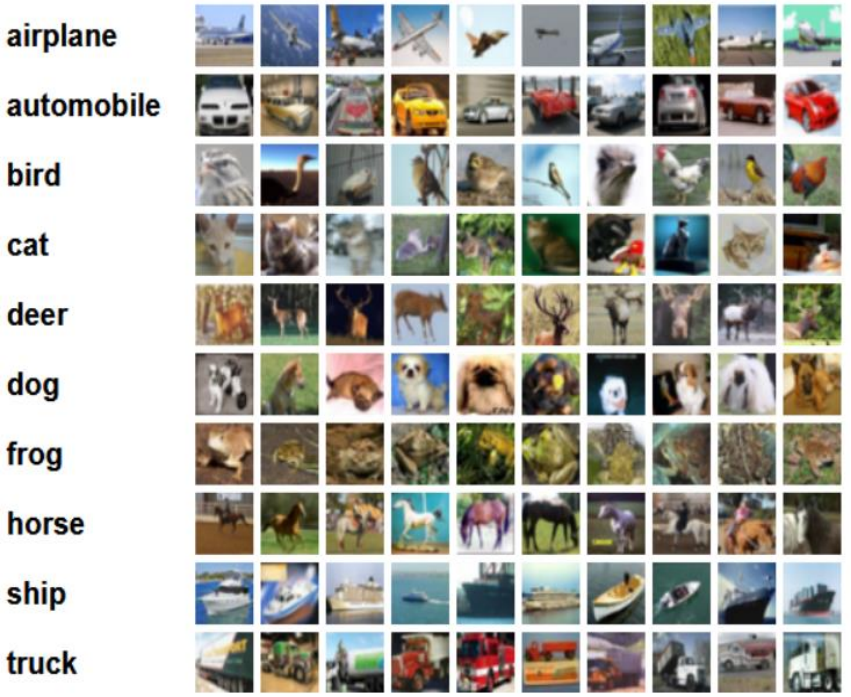




$$M_{op} = \sum_{i=0}^{n} (M_{ii} + M_{ik} + M_{ib}) + M_o$$

实验评估

- 实验环境
  - 开发板:思沃.r / SERVR.r
  - 测试模型: Resnet20
  - 数据集：Cifar10



实验评估

- 准确率
  - RVTensor和Keras的准确率一致

表 2 基于 resnet20 网络模型的准确率统计表

	Top1	Top5
RVTensor	77%	98%
Keras	77%	98%

★ Keras的准确性评估是基于X86平台完成的

- 性能
  - 处理每张图片的平均时间为13.51秒
- 执行文件大小
  - 193KB

未来工作

- 内存优化
  - SERVE.r的内存有限，推理过程中会有内存换入换出的开销
- 稀疏卷积优化
  - input输入数据中存在大量的零会导致卷积操作低效
- 模型剪枝
  - 通过剪枝技术，压缩模型参数，使其更加适合IoT应用场景
- V指令集适配
  - 基于V指令集重新实现op算子，提高算子执行效率

谢谢！

通讯作者：于佳耕(jiageng08@iscas.ac.cn)