

第六届

全国大学生集成电路创新创业大赛

报告类型\*： 测试报告

参赛杯赛\*： 海云捷讯杯

作品名称\*： 基于FPGA CNN加速器的SSD\_MobileNetV1

目标检测模型实现

队伍编号\*： CICC2707

团队名称\*： 十年踪迹十年芯

**目录**

[**1 模型概况 3**](#_Toc105247807)

[**1.1 模型概述 3**](#_Toc105247808)

[**1.2 16倍通道剪枝 4**](#_Toc105247809)

[**1.3 自创shadow模块 4**](#_Toc105247810)

[**1.4 CAM以及优化 5**](#_Toc105247811)

[**1.5 训练量化 7**](#_Toc105247812)

[**1.6 模型复杂度及性能 8**](#_Toc105247813)

[**2 推理测试 10**](#_Toc105247814)

[**2.1 精度测试 10**](#_Toc105247815)

[**2.2 推理速度 11**](#_Toc105247816)

[**2.3 FPGA推理结果 12**](#_Toc105247817)

[**3 优化模块消融实验 15**](#_Toc105247818)

[**3.1 16倍通道剪枝性能测试 15**](#_Toc105247819)

[**3.2 自创shadow模块性能测试 15**](#_Toc105247820)

[**3.3 通道注意力性能测试 15**](#_Toc105247821)

[**3.5 训练量化性能测试 16**](#_Toc105247822)

[**3.6 数据重排性能测试 18**](#_Toc105247823)

[**3.7 双缓冲性能测试 19**](#_Toc105247824)

# 1 模型概况

## 1.1 模型概述

本次设计模型优化采用了十六倍通道剪枝，在extra\_block运用了设计的shadow模块，引入了SE通道注意力，并采用训练量化的方法进一步加速。

模型整体的算子图示如下：

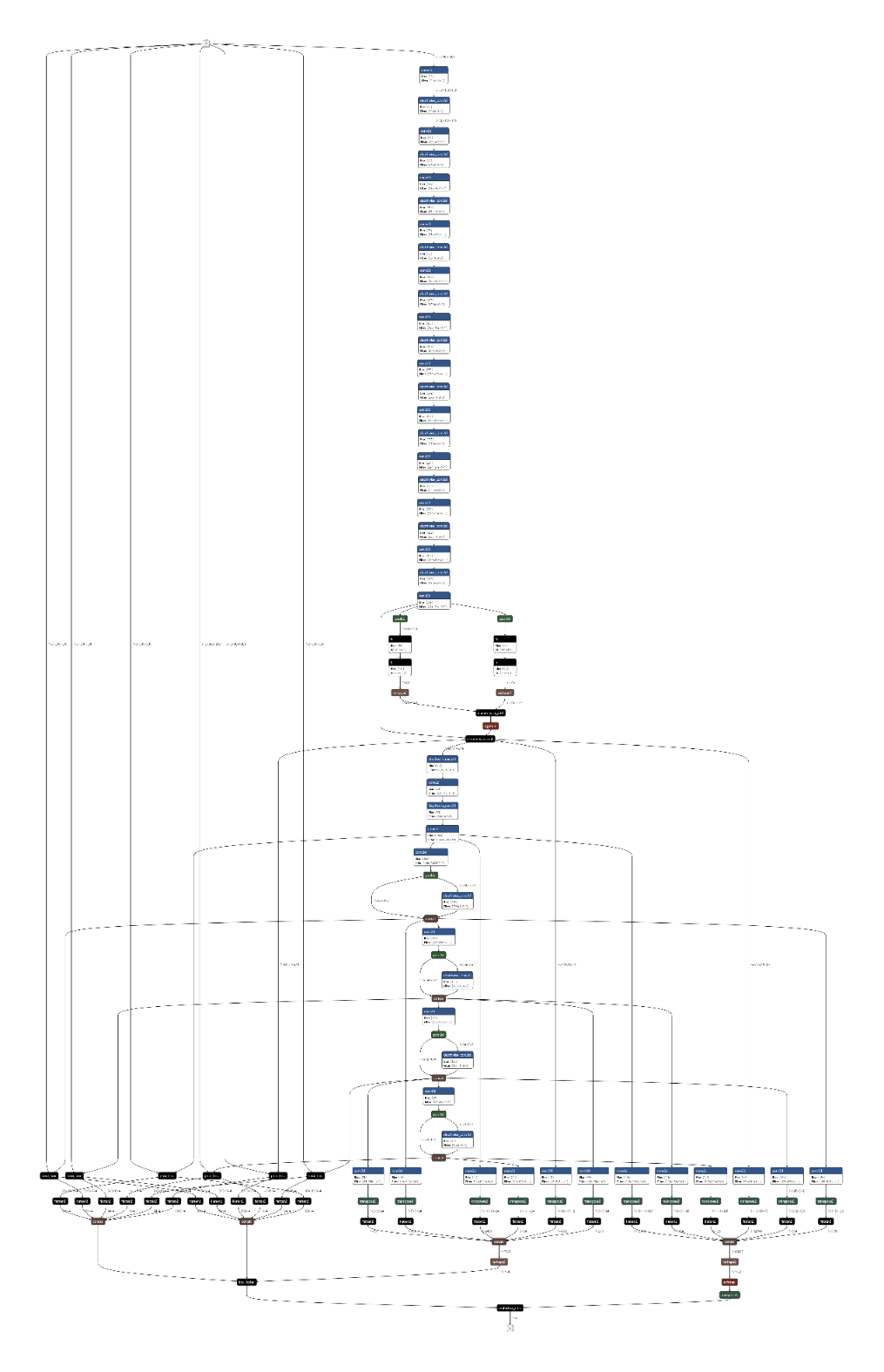


图1 arm版模型算子图

## 1.2 16倍通道剪枝

16倍通道剪枝后Mobilenet\_v1模型的结构如图所示：

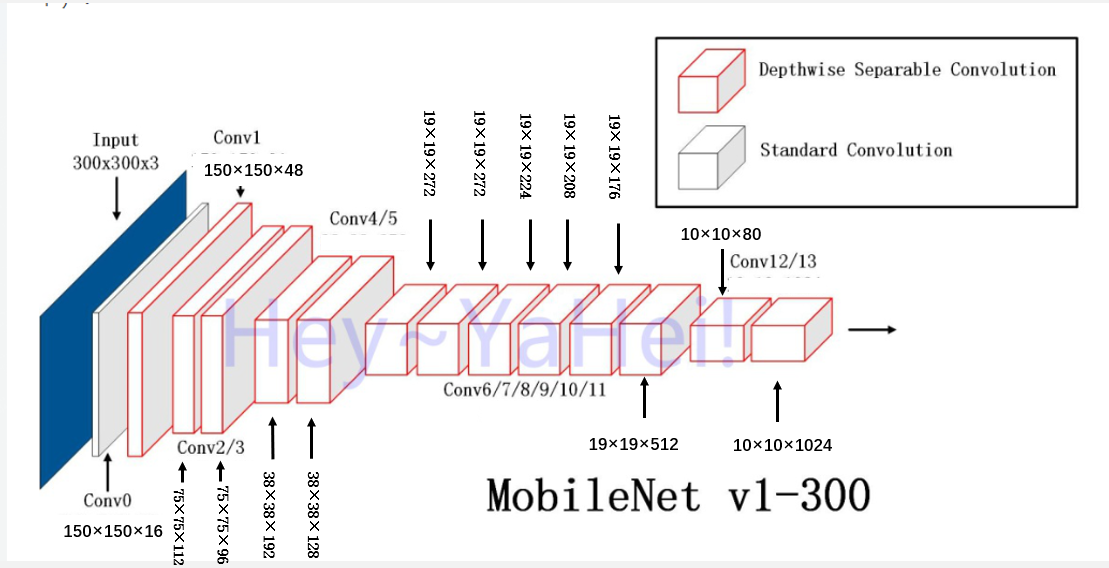


图2 mobilenet\_v1剪枝结构

## 1.3 自创shadow模块

在extra\_block引入的自创shadow模块算子结构如图所示：

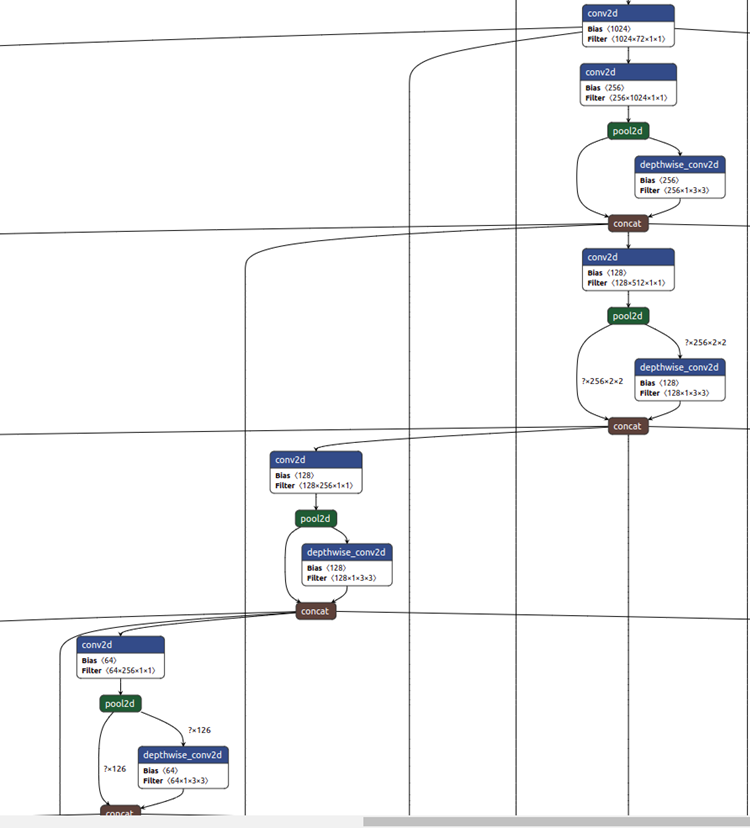


图3 shadow模块算子图

Shadow模块具有更精简的结构以及更加优越的推理性能。

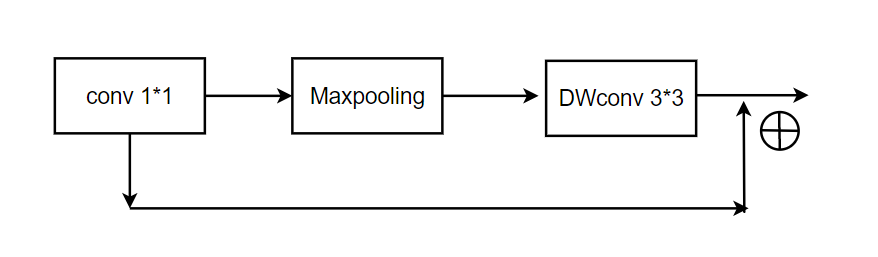


图4 shadow模块结构图

## 1.4 CAM以及优化

引入的CAM通道注意力模块算子结构如图所示：

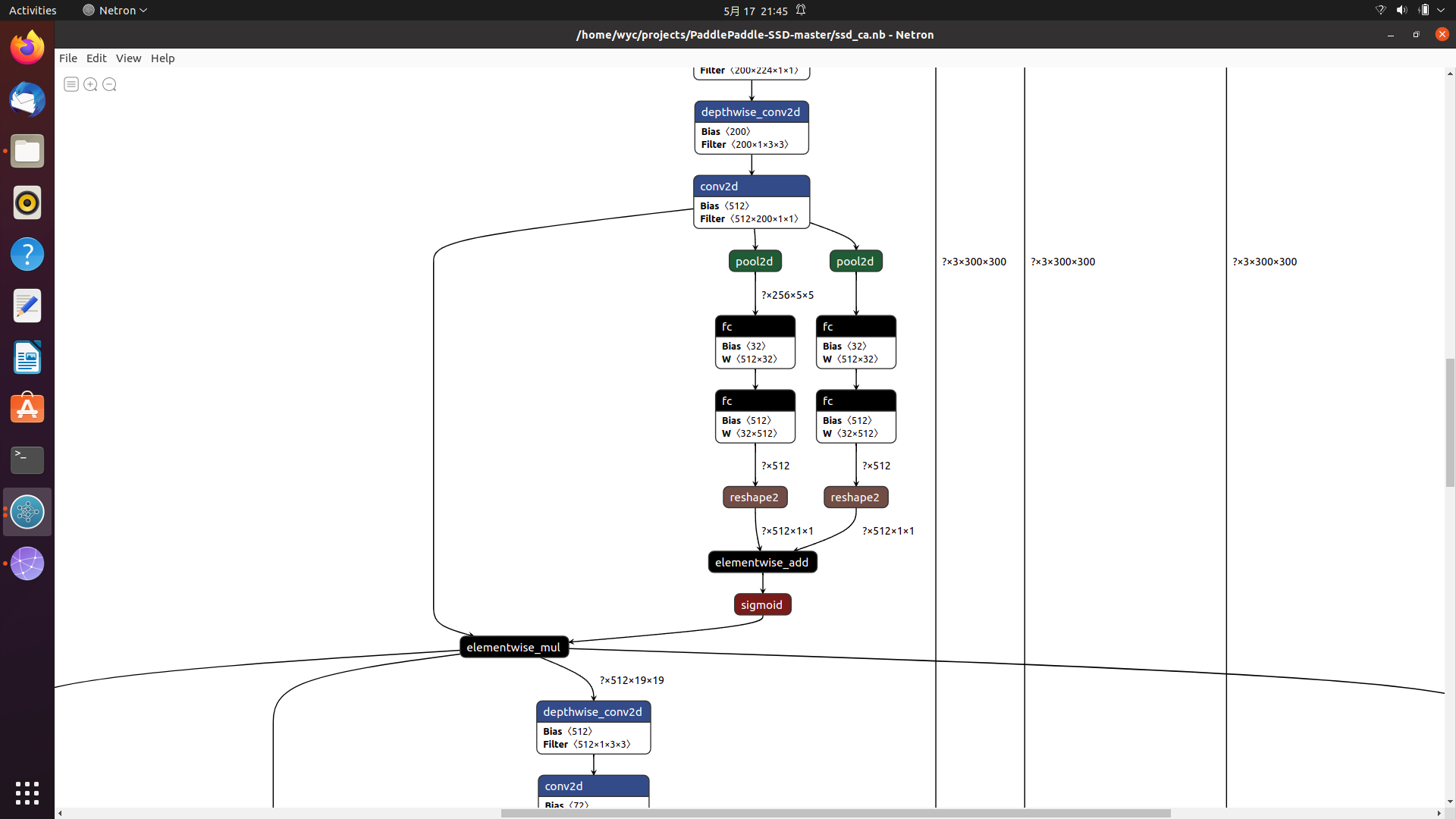


图5 CAM结构图

为了更进一步利用加速器的性能，实际采用二维1\*1的卷积替代CAM中的全连接层，利于加速器对卷积进行加速计算。

利用1\*1卷积替换全连接层之后，模型结构如图所示：

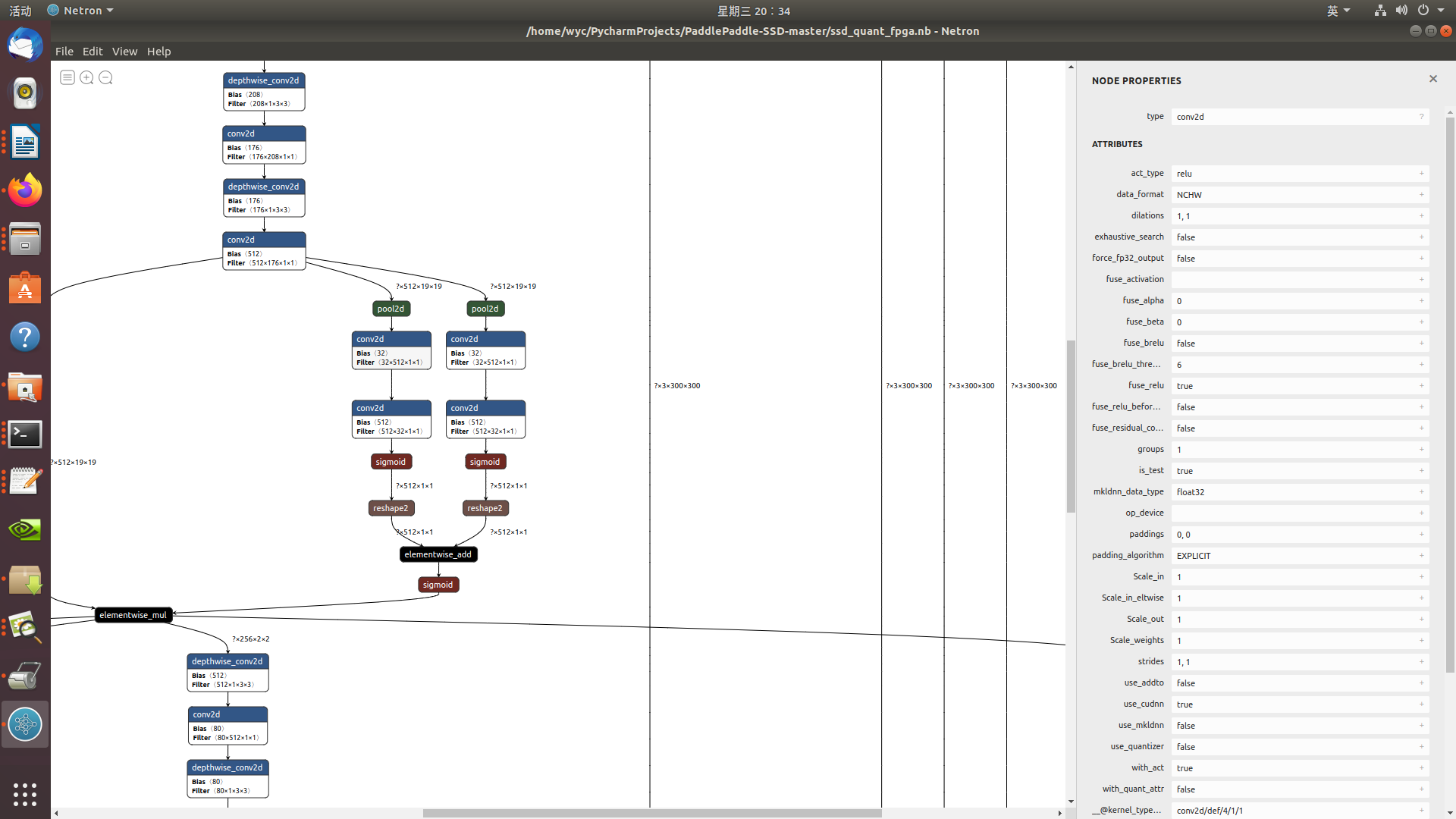


图6 优化后CAM结构图

## 1.5 训练量化

量化训练中，插入了量化反量化节点，以子图接入海云FPGA，模型算子图示如下：

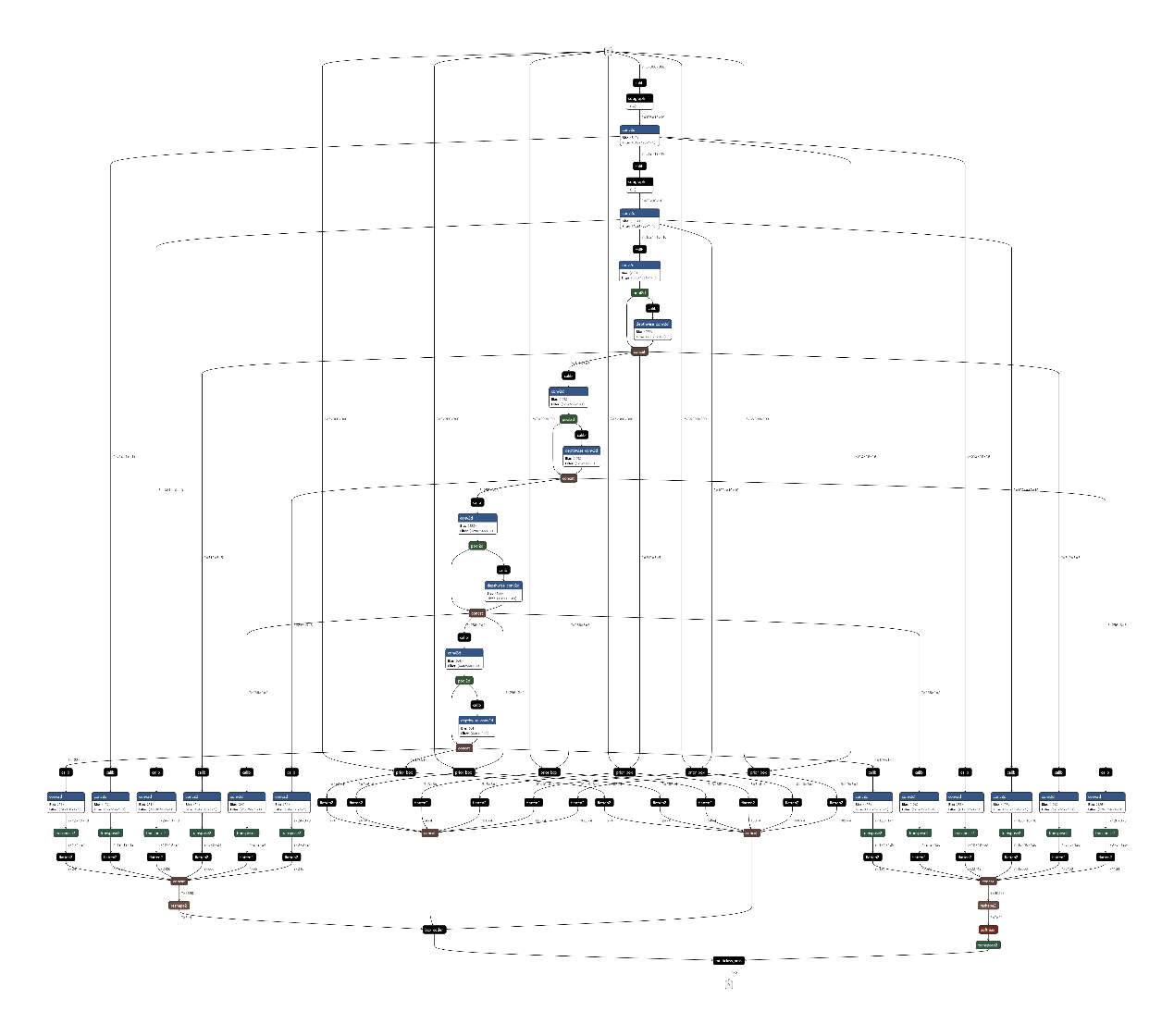


图7 arm+fpga版模型算子图

## 1.6 模型复杂度及性能

打印最终模型的结构参数信息：

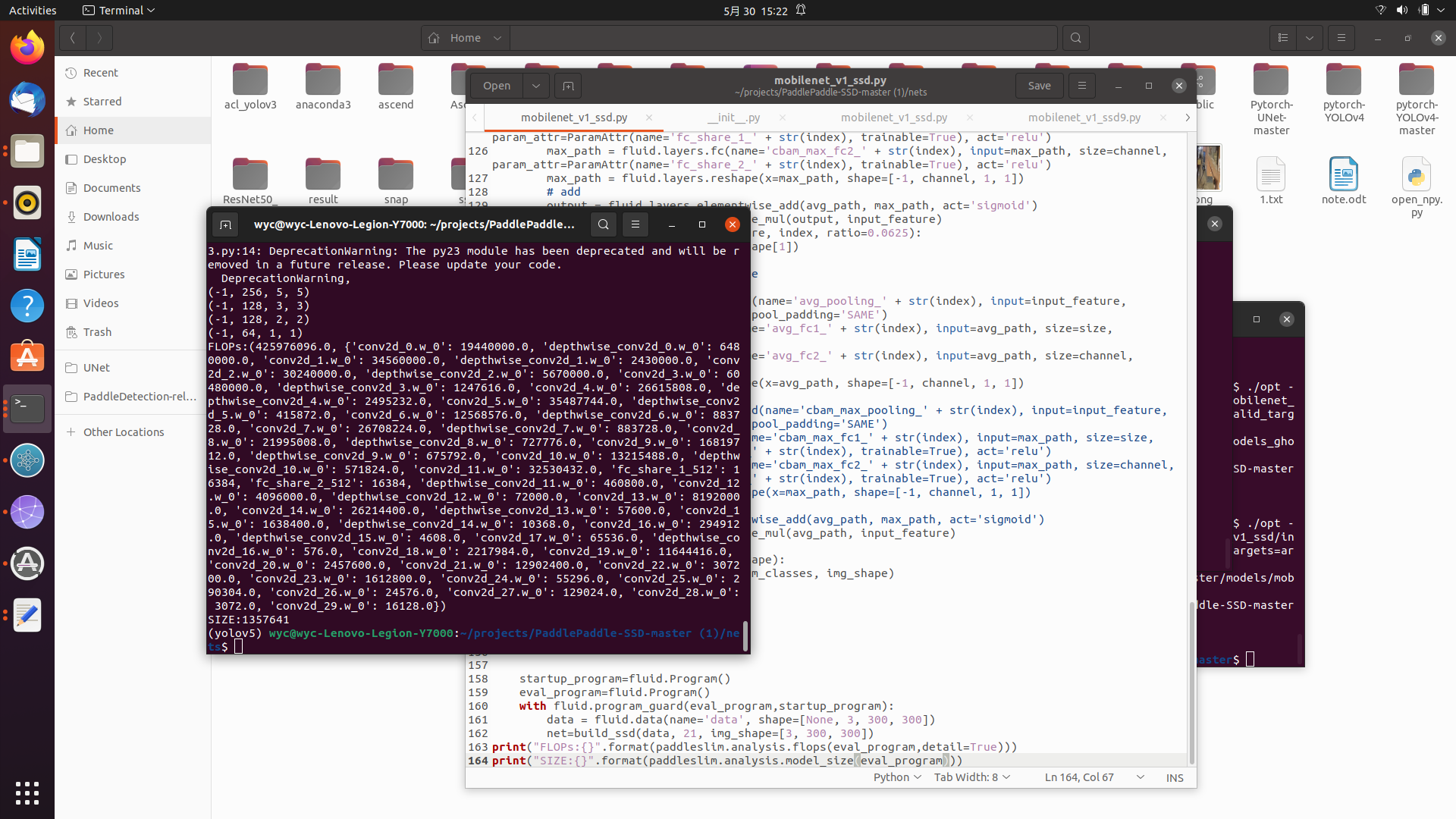


图8 模型的结构参数信息

对比优化前后模型的复杂度以及精度等信息。

表1 优化前后模型信息对比

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resuls** | FLOPs | **Parameters** | **Map** | **Inference Time(arm/fpga)** |
| Baseline | 1148479168 | 5821433 | 73.0 | 1100ms/1100ms |
| Modified | 425976096 | 1357641 | 65.0 | 218ms/500ms |

优化后模型参数量以及计算量明显减少，map能到到65，arm推理速度达四帧，arm+fpga推理速度达到两帧，符合设计要求。

# 2 推理测试

## 2.1 精度测试

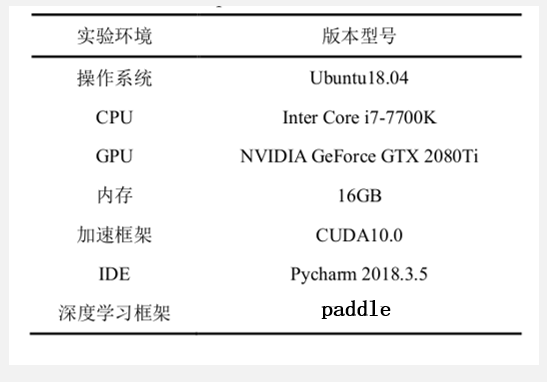


图9 训练环境

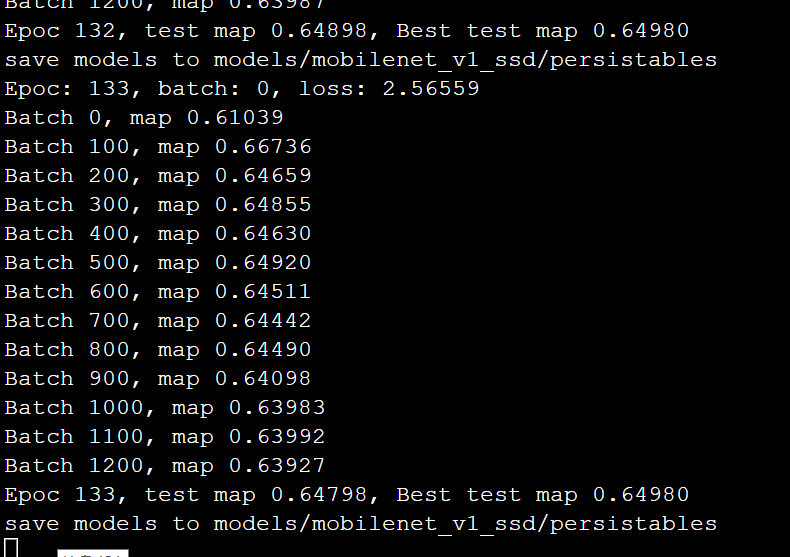


图10 模型精度测试

模型使用pascalvoc数据集进行训练，使用VOC2007测试集4952张图进行精度评估，达到了65的精度，符合设计要求。

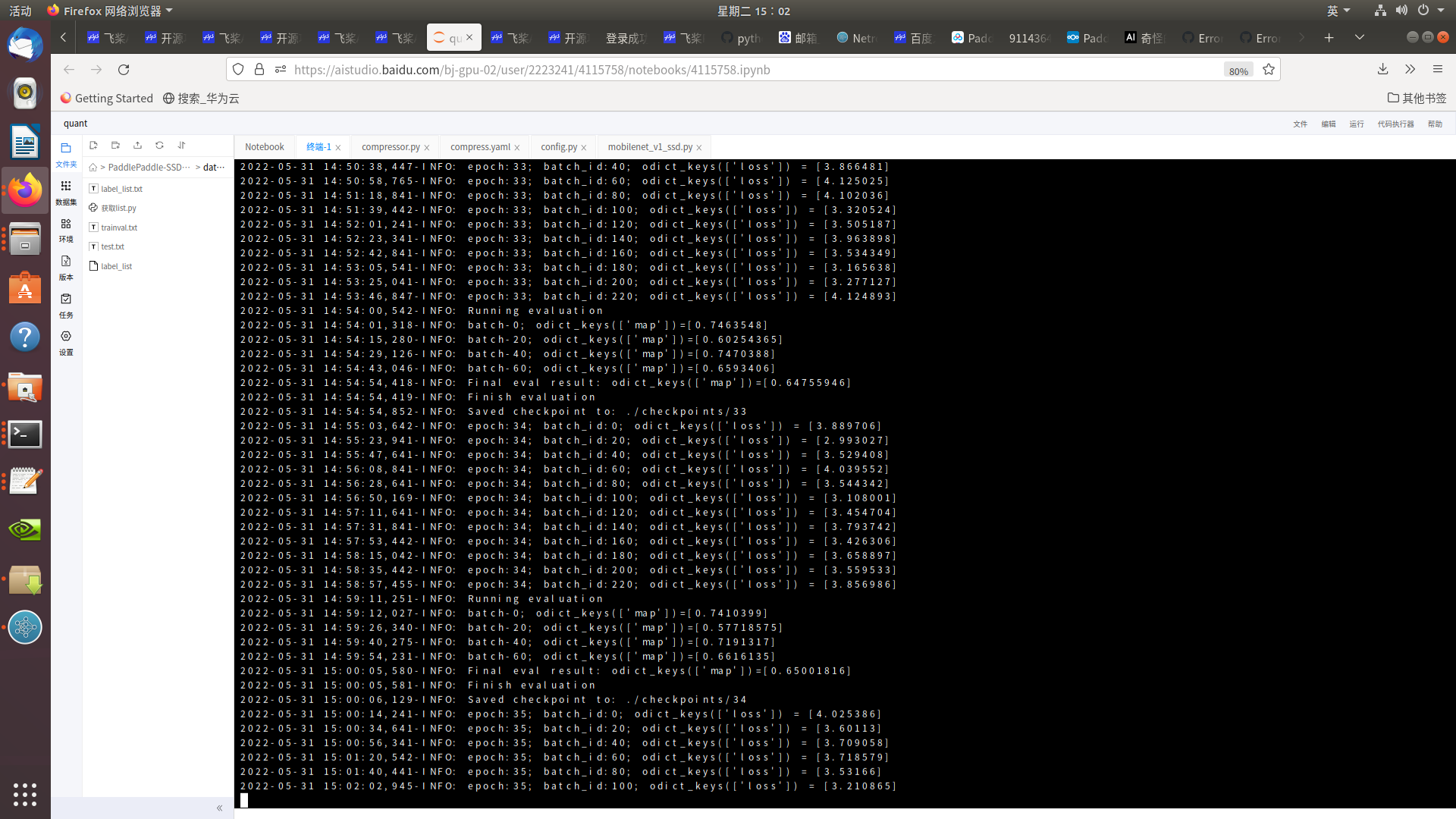


图11 量化训练精度测试

经过量化训练，同样使用VOC2007测试集4952张图进行精度评估，推理的精度达到了65以上，满足设计要求。

## 2.2 推理速度

在arm以及arm+fpga端进行模型推理测试。

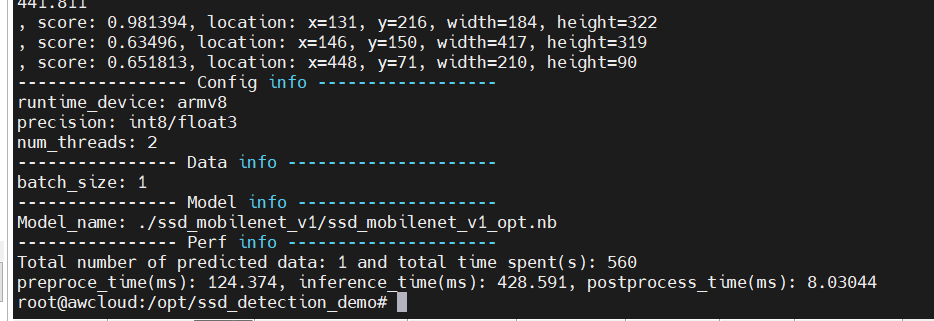


图12 arm推理速度

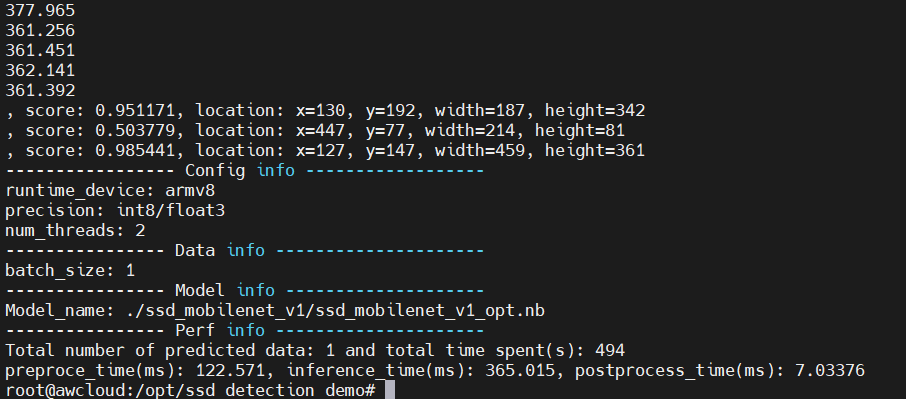


图13 arm+fpga推理速度

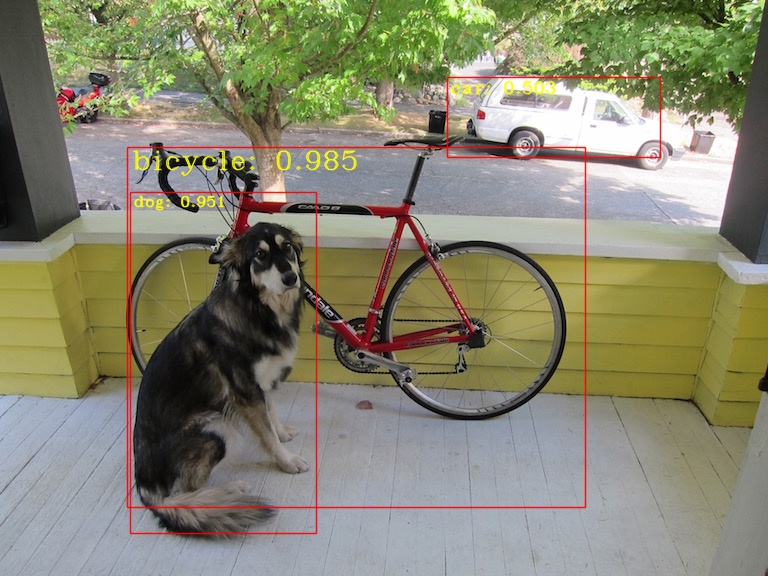


图14 推理结果

ARM端的推理速度，在各种优化后能够达到单张图片215ms，即四帧以上。ARM+FPGA端的推理速度，在各种优化后能够达到单张图片365ms左右，接近三帧，符合设计要求。

## 2.3 FPGA推理结果

测试图片共25张，推理结果如图所示。

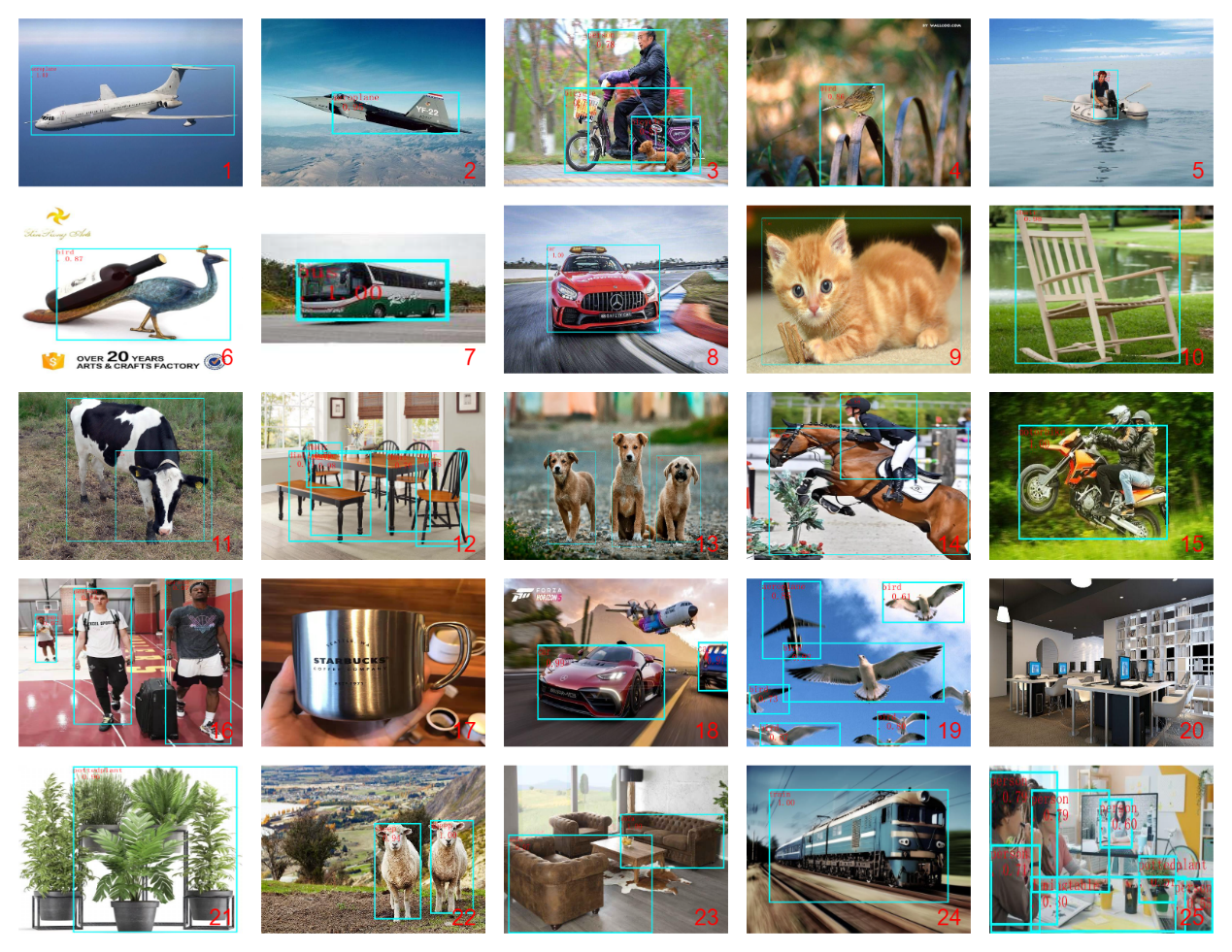


图15 测试集推理结果

用TP表示被模型预测为正类的正样本，TN表示被模型预测为负类的负样本，FP表示被模型预测为正类的负样本，FN表示被模型预测为负类的正样本。

设置IOU=0.5，统计25张图片分类正误情况。

表2 分类正误情况

|  |  |  |  |
| --- | --- | --- | --- |
| **Resuls** | TP | **FP** | **FN** |
| 1 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 |
| 3 | 2 | 1 | 1 |
| 4 | 1 | 0 | 0 |
| 5 | 2 | 0 | 0 |
| 6 | 1 | 3 | 1 |
| 7 | 1 | 0 | 0 |
| 8 | 1 | 0 | 0 |
| 9 | 1 | 0 | 0 |
| 10 | 1 | 0 | 0 |
| 11 | 1 | 1 | 0 |
| 12 | 4 | 0 | 2 |
| 13 | 1 | 3 | 2 |
| 14 | 0 | 0 | 2 |
| 15 | 1 | 0 | 2 |
| 16 | 3 | 0 | 1 |
| 17 | 1 | 0 | 0 |
| 18 | 2 | 0 | 1 |
| 19 | 3 | 0 | 5 |
| 20 | 0 | 0 | 2 |
| 21 | 5 | 1 | 1 |
| 22 | 2 | 0 | 0 |
| 23 | 2 | 0 | 2 |
| 24 | 1 | 0 | 0 |
| 25 | 2 | 1 | 3 |
| TOTOL | 39 | 10 | 26 |

计算精确度P=TP/(TP+FP)=39/49=0.796，计算召回率R=TP/(TP+FN)=39/65=0.6。

调整IOU的值，计算出测试集map。

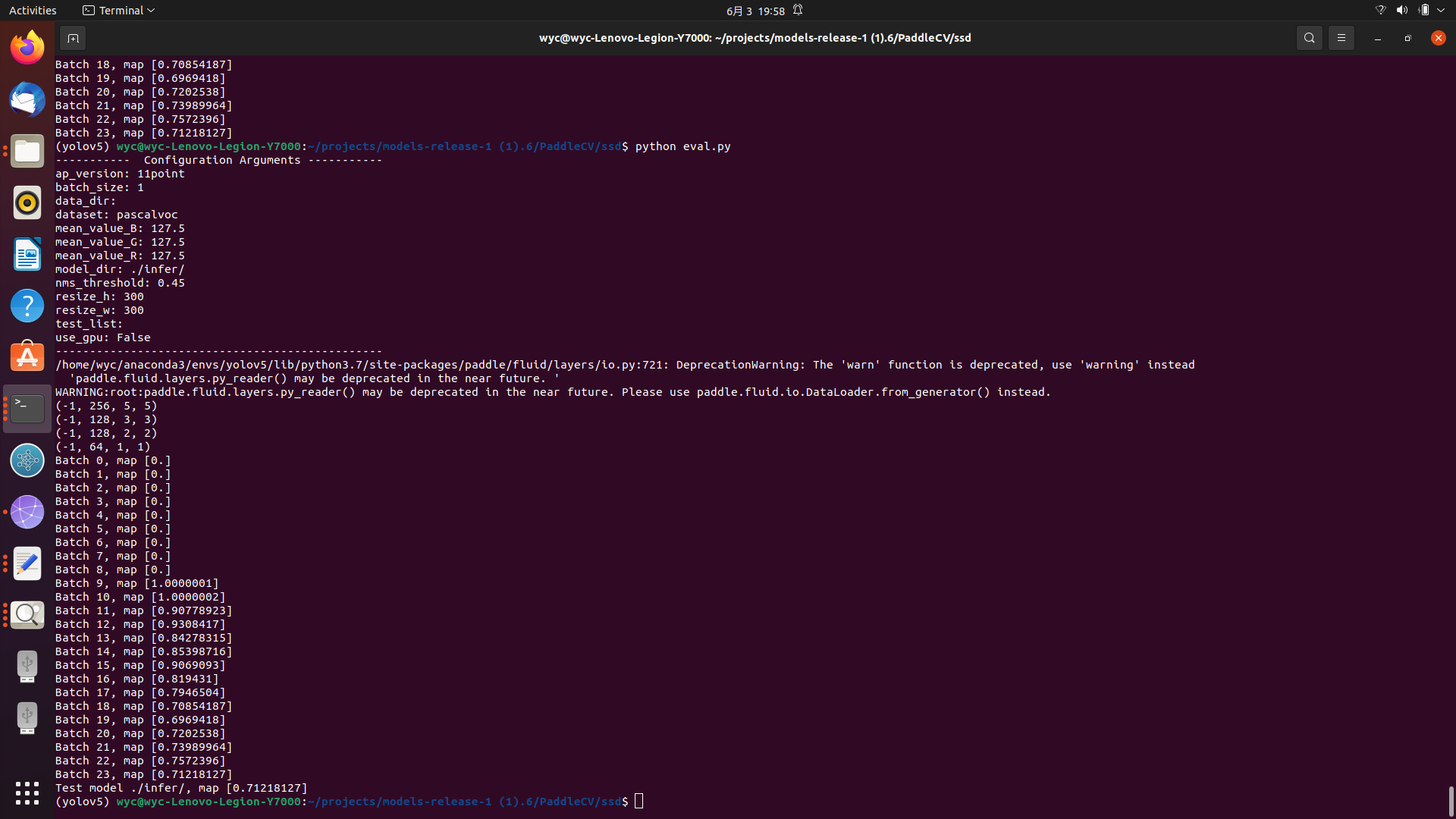


图16 25张测试集map结果

精度达到了0.71，远高于0.65，满足设计要求。

# 3 优化模块消融实验

## 3.1 16倍通道剪枝性能测试

对ssd\_mobilenet\_v1进行裁剪，对比裁剪前后的FLOPs以及参数量等信息。

表3 剪枝性能分析

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resuls** | FLOPs | **Parameters** | **Map** | **Inference Time(arm)** |
| Baseline | 1148479168 | 5821433 | 73.0 | 1100ms |
| Cut 60% | 449226444 | 3141669 | 65.5 | 550ms |
| 十六倍通道剪枝 | 459236192 | 3164105 | 66.0 | 544ms |

可以看出，在满足精度要求的前提下，设置剪枝比例为60%，模型的FLOPs以及参数量都下降至原来的一半左右，推理速度提升近一倍。所以，设置合适的剪枝比例进行训练至关重要。

而设计的16倍通道剪枝方法，由于剪枝后通道数量保持16的整数倍，可以在数据重排中，节省因为Padding而多出的计算量，与普通剪枝方法相比，具有很大的优势，因为普通剪枝方法剪枝后，通道数无法保证是16的整数倍，在数据重排中会有Padding操作，处理的数据总量变多。

## 3.2 自创shadow模块性能测试

在extra\_block区引入shadow模块，并对比原始的网络、ssdlite\_mobilenet\_v1以及引入shadow模块之后的网络的FLOPs以及参数量等信息。

表4 shadow模块性能分析

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resuls** | FLOPs | **Parameters** | **Map** | **Inference Time(arm)** |
| Baseline | 459236192 | 3164105 | 66.0 | 544ms |
| SSDlite | 446247776 | 2449609 | 64.0 | 460ms |
| Shadow | 425910560 | 1323785 | 64.5 | 420ms |

对比结果可以看出，shadow模块更加精简，其FLOPs以及参数量甚至都低于ssdlite\_mobilenet\_v1，拥有更快的推理速度，但是ssdlite\_mobilenet\_v1相比，引入shadow模块后，模型反而拥有更高的精度。推测因为最大池化保留了更加明显有用的特征，加上对通道的concat操作，合并了最大池化与DW卷积后的结果，起到了将深层低分辨率的精细特征与浅层高分辨率特征融合的效果，这种特征融合，能够弥补特征损失，提高模型精度。

## 3.3 通道注意力性能测试

在第一张512\*19\*19大小特征图前的引入通道注意力，并对比原始的网络、以及引入通道注意力模块之后的网络的FLOPs以及参数量等信息。

表5 通道注意力模块性能分析

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resuls** | FLOPs | **Parameters** | **Map** | **Inference Time(arm)** |
| Baseline | 425910560 | 1323785 | 64.5 | 420ms |
| SE | 425943328 | 1357097 | 64.6 | 430ms |
| CAM | 425976096 | 1357641 | 64.8 | 435ms |

付出很少的推理时间代价，使用CAM注意力，可以有效提升模型的检测精度。引入CAM注意力的模型，FLOPs和参数量的增加十分微小。

用卷积层替换全连接层后，重新对模型进行分析。

表6 CAM改进性能分析

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resuls** | FLOPs | **Parameters** | **Map** | **Inference Time(arm)** |
| CAM(fc) | 425976096 | 1357641 | 64.8 | 435ms |
| CAM(conv2d) | 425976096 | 1357641 | 65.0 | 435ms |

可以看出模型的参数量以及计算量没有变化，推理时间没有明显变化，但是精度略微提升，满足了设计要求。

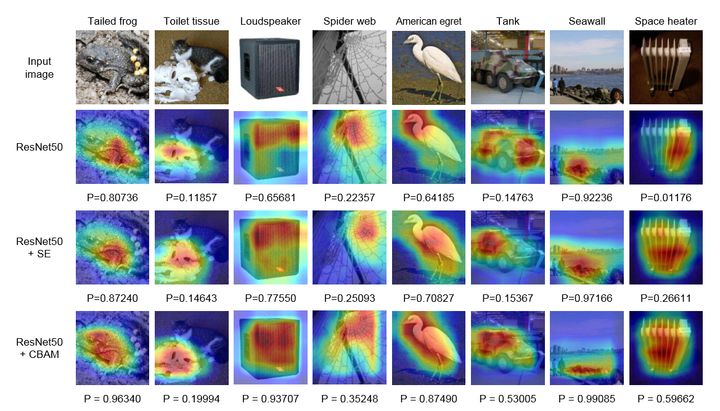


图17 CAM对注意力机制分析

利用 Grad-CAM 对不同的网络进行可视化后，可以发现，引入 CBAM 后，特征覆盖到了待识别物体的更多部位，并且最终判别物体的概率也更高，这表明注意力机制的确让网络学会了关注重点信息。

## 3.5 训练量化性能测试

使用以上模块，训练出满足精度要求的模型，再进行量化训练，比较arm以及fpga端的推理结果。

表7 训练量化性能分析

|  |  |  |  |
| --- | --- | --- | --- |
| **Resuls** | Size | Map | **Inference Time(arm/fpga)** |
| Baseline | 5.6M | 65.0 | 435ms/435ms |
| 量化训练 | 1.6M | 65.0 | 215ms/521ms |

可以看出，经过训练量化后，在RAM端的推理速度有明显提升。但是FPGA端的推理速度下降了，可能的原因是FPGA端的优化，例如双缓冲、深度可分离卷积算法的优化、ko驱动编写没有到位。

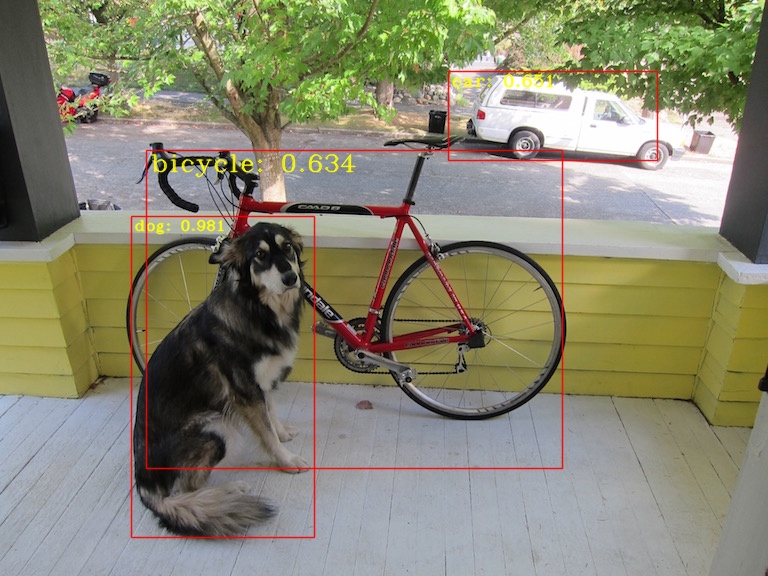


图18 全精度模型推理结果

ARM端的推理速度，在各种优化后能够达到单张图片215ms，即四帧以上。ARM+FPGA端的推理速度，在各种优化后能够达到单张图片500ms左右，即两帧，符合设计要求。

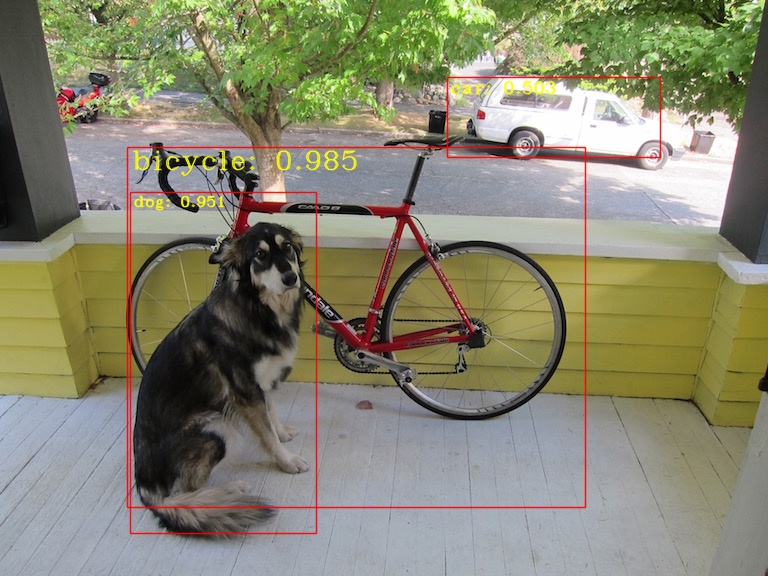


图19 量化训练模型推理结果

由实验可知，量化训练模型推理精度并没有明显下降，鲁棒性很好。

## 3.6 数据重排性能测试

重新用neon编写utils.cc的数据输入以及输出重排和数据拷贝函数，并删减不必要的内存拷贝，测得推理性大幅提升，实测能够达到近3帧的水平。

表9 数据重排性能分析

|  |  |
| --- | --- |
| **Resuls** | **Inference Time(fpga)** |
| Baseline | 521ms |
| 输入重排 | 518ms |
| 输出+输入重排 | 515ms |
| 数据重排+neon拷贝 | 512ms |
| 数据重排+neon拷贝+减少拷贝 | 365ms |

## 3.7 双缓冲性能测试

在驱动计算刚开始时，将下一节点的权重和fpga参数包可以预先一次性传输到指定的DDR空间中，利用计算时间来掩盖数据传输时间，测得推理性能有5ms的提升。

表10 数据重排性能分析

|  |  |
| --- | --- |
| **Resuls** | **Inference Time(fpga)** |
| Baseline | 365ms |
| 双缓冲 | 360ms |