# 基于通道裁剪的YOLOv3模型

毛雪宇[1,2]，彭艳兵[2]
（1.武汉邮电科学研究院 湖北 武汉 430074；2.南京烽火天地通信科技有限公司 江苏 南京 210019）

**摘要：**为进一步加速基于深度学习的目标检测模型，采用通道裁剪的方法对YOLOv3模型进行瘦身。首先选取卷积神经网络模型中可以裁剪的卷积层，通过对BN层参数进行稀疏化训练；其次，对BN层参数排序以获得可以裁剪的通道；然后依据参数范围裁剪模型；最后对裁剪后的模型进行微调。在X光图像数据集上进行试验，瘦身后的模型mAP@0.5值提高2%，预测过程提速16.67%。实验证明，在X光检测任务中，裁剪后的YOLOv3模型依然可以减少部分过拟合现象并提高准确率和预测速度，保证了本文模型的实用性。

**关键词：**目标检测；通道裁剪；YOLOv3；卷积神经网络；深度学习
**中图分类号：**TP391.4 **文献标识码：**A **文章编号：**1674-6236(2020)16-0137-05
DOI：10.14022/j.issn1674-6236.2020.16.030

## YOLOv3 based on network slimming

MAO Xue-yu[1,2], PENG Yan-bing[2]
(1. *Wuhan Research Institute of Posts and Telecommunications*, *Wuhan* 430074, *China*; 2. *Nanjing FiberHome World Communication Technology Co., Ltd.*, *Nanjing* 210019, *China*)

**Abstract:** In order to further accelerate the object detection model based on deep learning, this paper uses channel pruning method to slim the weight of YOLOv3 model. Firstly, the convolution layer which can be pruned in the convolution neural network is selected, then the parameters of BN layer are sparsely trained; secondly, the parameters of BN layer are sorted to obtain the channels that can be pruned; thirdly, the model is pruned according to the parameters range; finally, the pruned model is fine-tuned. Experiments on the X-ray image datasets shows that the mAP@0.5 value of the model increased by 2% and the prediction process increased by 16.67%. Experiments also show that in X-ray detection tasks, the pruned YOLOv3 model can still reduce some over-fitting phenomena and improve the accuracy and prediction speed, thus ensuring the practicability of this model.

**Key words:** object detection; channel prune; YOLOv3; convolutional neural network; deep learning

近几年来，卷积神经网络(Convolutional Neural Networks, CNN)已经广泛应用于多种计算机视觉任务中，如图像识别、图像分割[1]和目标检测等[2]。从AlexNet、VggNet和GoogleNet到ResNet等，模型层数已经从8层发展到100层以上。更大的CNN虽然具有更强的特征表达力，却更需要资源。152层ResNet具有超过6000万个参数，在资源受限的平台，如移动设备、可穿戴设备或物联网设备上，不可能负担得起，于是模型压缩成为了研究热点。同时，Faster-RCNN等两阶段目标检测模型耗时太长，工业级场景中YOLOv3(You Only Look Once version 3)[3]则更为常用，但是其预测速度仍然可以依据训练数据有相应的提升空间。本文针对以上问题，将通道裁剪方法在YOLOv3目标检测模型上进行实验。

## 1 相关研究

自2013年R-CNN模型首次提出，目标检测在当

**收稿日期：**2019-08-22 **稿件编号：**201908115
**基金项目：**国家重点研发计划“智能服务交易与监管技术研究”项目(2017YFB1400704)
**作者简介：**毛雪宇(1995—)，男，江苏淮安人，硕士研究生。研究方向：图像检索和目标检测。

前公开数据集如COCO和VOC上已经取得了显著的提升，当前速度与准确率较优的目标检测模型有CenterNet[4]，YOLOv3和RefineDet，参考图1。其中，CenterNet训练难度大，模型不易收敛，而RefineDet开源方法中使用caffe框架，对于使用者改进网络难度较大。

| 模型 | 输入图像大小 | 主干网 | mAP @0.5(%) COCO | FPS |
|---|---|---|---|---|
| RefineDet | 320*320 | VGG-16 | 49.2 | 40.3 |
| | 512*512 | VGG-16 | 54.5 | 24.1 |
| | multiscale | res-101 | 62.9 | - |
| CenterNet | 512*512 | DLA-34 | 57 | 52 |
| | 512*512 | res-101 | 53 | 45 |
| | 512*512 | Hour-104 | 59.1 | 14 |
| YOLOv3 | 320*320 | darknet-53 | 51.5 | 45 |
| | 416*416 | darknet-53 | 55.3 | 35 |
| | 608*608 | darknet-53 | 57.9 | 20 |
| | 416*416 | tiny | 33.1 | 220 |
| | 608*608 | spp | 60.6 | 20 |

图1　模型性能对比

### 1.1　YOLOv3模型结构

YOLOv3是YOLO系列模型中首次引入残差网络[5]，其网络结构如图2所示。

对于第二个版本YOLO9000使用的是DarkNet-19，一种类似于VGG的网络。残差网络结构残差结构还有助于模型梯度的平缓下降，进一步提升了网络的深度，提高了模型的准确率[6]。

除残差块的使用外，YOLOv3中参考SSD[7]等方法引入多尺度特征图以提高目标检测的精度[8]。一共有3种尺度的特征图，对于每一种尺度的特征图，分别有3组锚框与之对应[9]。实测效果证明该方法确实有效提高小目标物体的召回率，解决了前两个YOLO版本中的缺陷。

### 1.2　模型压缩与加速方法

文献[10]中提出一种对全连接层参数使用奇异值分解的方法来获得全连接参数的低秩矩阵，从而完成模型压缩，但是此类方法已经不再适用于当前的深度CNN卷积网络，因为ResNet等网络并不是由全连接层组成的网络。

Han等[11]使用一种改进量化的方法来完成对模型的压缩，但是该方法在运行时任然需要将权重还原回原始卷积核，因此并不能节省时间和运行显存的使用。Rastegari[12]将模型权重量化成{-1，1}或者{-1，0，1}，此类方法需要特定的硬件加速库才能完成，与当前PyTotch / TensorFlow等框架不能兼容。

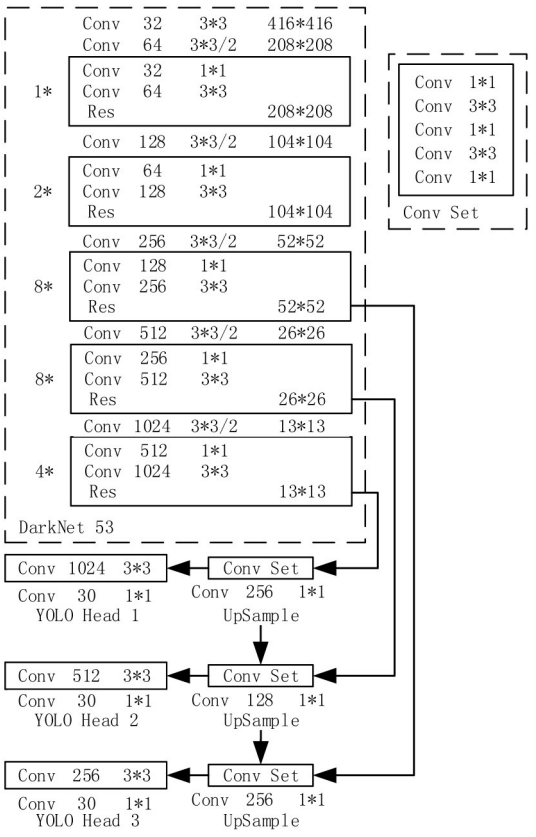


图2　5类别416*416 YOLOv3结构

文献[13]中认为模型结构是模型效果重要的影响参数，且通过实验来验证该观点，但是对于模型结构在什么情况下最优并没有给出参考。通过网络结构自动探索（NAS[14]）等方法可以给出最优结构，但是通常需要消耗大量算力来完成。

CNN中卷积层的权值为4维张量，因此对它的剪枝按剪枝粒度的粗细可分为4类，从最细粒度的单个权值剪枝到最粗粒度的通道剪枝[15]。基于通道裁剪的模型瘦身方法在[16]和[17]中提到，该方法在卷积核通道上进行裁剪。对于大多数深度学习框架，在构建模型的时候就需要指定卷积核参数的大小，其中包含通道数，因此不需要特殊的硬件加速库或者辅助工具。对于模型的要求在于模型需要有批归一化层（Batch Normalization，BN），对VGG等不含有BN层的模型并不适用。

## 2　YOLOv3通道裁剪

### 2.1　通道裁剪原理

#### 2.1.1　BN层原理

在CNN模型中，BN层对一个mini-batch[18]进行

归一化,通常放置于卷积层后,激活层前。对于每一个输入 $z_{in}$ ,BN层输出 $z_{out}$ 之间有:

$$\hat{z}=\frac{z_{in}-\mu_B}{\sqrt{\sigma_B^2+\varepsilon}} \qquad (1)$$
$$z_{out}=\gamma\hat{z}+\beta$$

其中 $B$ 表示当前的mini-batch。从式(1)中不难看出BN层的作用相当于去均值除标准差并缩放加偏置操作。该操作以通道为基础,在每一个通道上对mini-batch和特征图进行统计并计算均值和方差。BN层的参数 $\gamma$ 和 $\beta$ 都是一组以通道数为其长度的向量。

### 2.1.2　BN层稀疏化与裁剪

BN层稀疏化是模型通道裁剪的关键,裁剪时通过BN层参数 $\gamma$ 来判断哪些通道对下一层影响不大,哪些通道是下一层重要的输入来源,如图3所示。

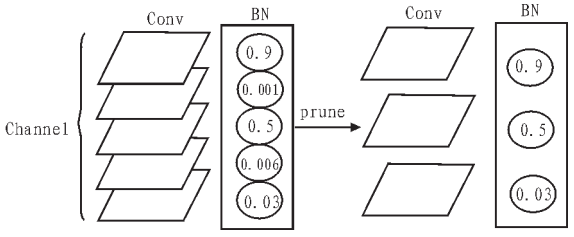


图3　BN层稀疏化与裁剪

稀疏化可以用L1正则化来完成,于是整体损失函数变为:

$$L=\sum_{(x,y)}l\left(f(x,W),y\right)+\sum_{\gamma\in\Gamma}|\gamma| \qquad (2)$$

其中 $(x,W,y)$ 分别表示输入数据,模型可训练权重和目标数据,公式第一项表示正常的目标检测模型损失,第二项表示加在BN层上的L1正则损失项,$\Gamma$ 表示所有的BN层参数。

### 2.1.3　裁剪流程

整体裁剪流程如图4所示,包含稀疏化训练/裁剪和微调3步。

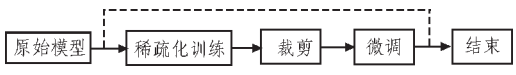


图4　裁剪流程

其中,多次稀疏化至微调的过程能够进一步压缩CNN模型。

### 2.2　可裁剪通道

由于BN稀疏化的过程中无法保证每一层可裁剪通道数可预计,因此对于含有特殊结构的模型,如残差结构,如图2所示。该结构中需要注意的是在跳连所连接的两个卷积层需要保证裁剪后通道数相同,否则会导致跳连两端维度不同,无法相加,从而前向传播失败。

针对这种情况,文献[16]中提出的原始方法是对于跳连的两端不再进行裁剪,保留其通道数,本文也将遵循该思路。

在DarkNet-53中,模型始终是每两层中第一层的输入和第二层输出进行跳连,因此裁剪第一层,如图5所示。

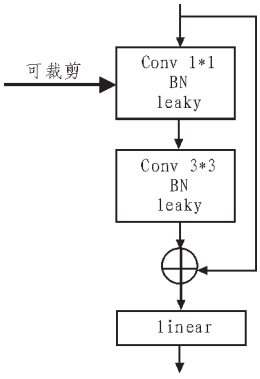


图5　YOLOv3模型结构

对于模型最后三层不同尺度的输出层,每一层包含类别的概率估计和边框参数回归值,该类卷积层每一个输出通道都有其特定的任务,因此也不能裁剪。

## 3　实验结果

### 3.1　实验数据集与数据预处理

X光限制品数据集由阿里天池比赛津南数字制造算法挑战赛公布,共有5类限制品,包含塑料打火机、铁壳打火机、水果刀、电池和剪刀。数据集中有5 420张图像包含限制品,2 000张图像为纯背景图,即图像中不包含限制品。数据集共有14 401个标注框,塑料打火机4 359个,电池5 254个,此两类数量最多且物体较小,铁壳打火机、剪刀和刀具数量小于2 000。

预处理中对不包含限制品的图像进行贴图处理,使纯背景图像中至少包含一类限制品。图像宽度从156像素到884像素不等。

对于处理后的含标签图像数据,随机抽取10%作为测试集,5%为验证集,其余图像用于训练。

### 3.2　实验设置

#### 3.2.1　实验环境

本实验在E5 2689 v4双核128G内存服务器上运行,使用一张1080Ti 11G显存显卡进行训练和裁剪。

#### 3.2.2 测试指标

实验参考Pascal VOC数据集，测试交占比（IOU）大于0.5时的平均准确率（mAP@0.5），召回率recall，准确率precision和f1值。

#### 3.2.3 训练细节

模型使用416*416的图像作为模型输入，稀疏化训练时，稀疏化参数 $s$ 为0.01，学习率为0.001，5轮更新一次学习率为原来的0.1倍，初始权重为YOLOv3预训练权重，batch-size为16，训练至损失不降。

此时有50%的BN层通道参数小于1e-4，因此裁剪掉了50%的权重。

在微调时，以0.000 1学习率微调5轮。

### 3.3 结果对比

对可裁剪的卷积层的BN层参数进行排序，划分10个区间，取区间里的最大值记录在表1中。

表1 稀疏化训练BN层分布

| BN层区间 | 区间最大值 |
|---|---|
| 0~10% | 6.7611e-6 |
| 10~20% | 1.4347e-5 |
| 20%~30% | 2.4326e-5 |
| 30%~40% | 4.0021e-5 |
| 40%~50% | 6.7408e-5 |
| 50%~60% | 0.000112 |
| 60%~70% | 0.000217 |
| 70%~80% | 0.000526 |
| 80%~90% | 0.017317 |
| 90%~100% | 0.590426 |

表1展示了稀疏化训练结束后BN层参数情况，其中有50%的参数小于1e-4，80%的参数小于1e-3。

表2对比裁剪前YOLOv3结果与裁剪50%后模型结果，从表中不难看出，当前模型在裁剪微调后，召回率降低约3%，准确率提升约5%，mAP提升约2%。

表2 裁剪前后结果对比

| 模型 | 准确率 | 召回率 | mAP | f1 |
|---|---|---|---|---|
| 裁剪前 | 0.794 4 | 0.912 3 | 0.862 0 | 0.849 2 |
| 50%裁剪 | 0.849 8 | 0.880 7 | 0.887 7 | 0.863 3 |

表3为通道裁剪后的模型结构表，其中对于不能裁剪的层如跳连层，即Conv3、Conv5等，都没有进行裁剪，其输出通道数不变。

模型测试时，平均预测时间从0.018 s提高至0.015 s，预测时间上整体提升16.67%。

### 3.4 结果分析

对于裁剪后的模型，召回率降低准确率提高[19-21]，即裁剪的过程降低部分过拟合的现象。对此，分析如下，首先模型裁剪的时候将一些不必要的参数丢弃，比如BN层参数低于1e-4所对应的通道，使模型权重能够更加集中于有效的目标物体。其次，在微调的时候模型结构已经固定，模型参数量无法增加，使得模型的拟合能力受到限制，只能在有限的空间内进行调整，从而避免了过拟合的现象，因此降低了模型的召回率，提升了准确率，但整体上 $f1$ 值依旧有提升。

表3 模型裁剪后通道情况

| Conv | 原输出 | 现输出 | 原输入 | 现输入 |
|---|---|---|---|---|
| 1 | 32 | 30 | 3 | 3 |
| 2 | 64 | 64 | 32 | 30 |
| 3 | 32 | 20 | 64 | 64 |
| 4 | 64 | 64 | 32 | 20 |
| 5 | 128 | 128 | 64 | 64 |
| 6 | 128 | 26 | 128 | 128 |
| 7 | 128 | 128 | 128 | 26 |
| 8 | 64 | 49 | 128 | 128 |
| …… | …… | …… | …… | …… |

预测速度的提升是进行模型瘦身的主要目标。由于模型通道数的减少，参与运算的参数相应减少，预测速度自然会有提升。

## 4 结 论

对于深度卷积神经模型，如何选用合适的网络模型来解决当前的问题一直是研究者们关心的问题。对于通道裁剪，可以认为是通过设定规则让模型去自动探索最优通道，然后人工裁剪。相对于AutoML等自动调优方法，通道裁剪不需要去重新搜索模型合适的结构，而是在已有的模型结构上进行调优，只需要正常的训练且对于特定的数据可以找出其特有的通道结构，因此节省了大量的运算资源。

但是该方法对于one-shot-learning或few-shot-learning等少样本学习应用场景，通道裁剪的方法并不适用。首先样本数量不足则不能通过长期训练使模型BN层稀疏；其次，少样本学习依靠的是基础模型的泛化能力来达到识别或分类的目的，裁剪后的模型泛化能力因其参数的减少必然有所欠缺；最后，对于特殊物体的目标检测应用场景，少样本学习本身就不适用。

对于当前X光数据，不在常用数据集类别中，且与常用数据集有较大的差异，无论是颜色信息还是待检测的物品，都不是上述少样本的应用场景，因此

可以通过设定损失函数对YOLOv3的BN层进行稀疏来探索合适的模型结构。最后得到了合适的模型通道数,以此来达到模型瘦身的目的。对于X光模型的训练过程,还可以添加如随机翻转、随机缩放等变化来进一步提升模型的抗干扰能力。

**参考文献:**


[1] Badrinarayanan V, Kendall A, Cipolla R, et al. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation[J].IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495.

[2] 焦李成,杨淑媛,刘芳,等.神经网络七十年: 回顾与展望[J].计算机学报,2016,39(8): 1697-1716.

[3] Redmon J, Farhadi A. Yolov3: An incremental improvement[EB/OL].(2018) [2019-08-07] arXiv preprint arXiv:1804.02767.

[4] Duan K, Bai S, Xie L, et al.CenterNet: Object Detection with Keypoint Triplets[EB/OL]. (2019) [2019-08-07] arXiv preprint arXiv:1904.08189.

[5] 张索非,冯烨,吴晓富.基于深度卷积神经网络的目标检测算法进展[J].南京邮电大学学报:自然科学版,2019,39(5):72-80.

[6] Montavon G, Samek W, Muller K, et al. Methods for Interpreting and Understanding Deep Neural Networks[J].Digital Signal Processing, 2018: 1-15.

[7] 张泽苗,霍欢,赵冯禹,等.深层卷积神经网络的目标检测算法综述[J].小型微型计算机系统,2019,40(9):1825-1831.

[8] 陈幻杰,王琦琦,杨国威,等.多尺度卷积特征融合的SSD目标检测算法[J].计算机科学与探索,2019, 13(6): 1049-1061.

[9] 陈聪,杨忠,宋佳蓉,等.一种改进的卷积神经网络行人识别方法[J].应用科技, 2019, 46(3): 51-57.

[10]Denton E L, Zaremba W, Bruna J, et al. Exploiting linear structure within convolutional networks for efficient evaluation[C]//Advances in neural information processing systems.2014: 1269-1277.

[11]Han S, Pool J, Tran J, et al. Learning both weights and connections for efficient neural network[C]// Advances in neural information processing systems. 2015: 1135-1143.

[12]Rastegari M, Ordonez V, Redmon J, et al. Xnor-net: Imagenet classification using binary convolutional neural networks[C]//European Conference on Computer Vision. Springer, Cham, 2016: 525-542.

[13]Liu Z, Sun M, Zhou T, et al. Rethinking the value of network pruning[EB/OL].(2018) [2019-08-07] arXiv preprint arXiv:1810.05270.

[14]Elsken T, Metzen J H, Hutter F.Neural Architecture Search: A Survey[J].Journal of Machine Learning Research,2019,20(55): 1-21.

[15]姜春晖.深度神经网络剪枝方法研究[D].合肥: 中国科学技术大学, 2018.

[16]Liu Z, Li J, Shen Z, et al. Learning efficient convolutional networks through network slimming [C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 2736-2744.

[17]He Y, Zhang X, Sun J, et al. Channel Pruning for Accelerating Very Deep Neural Networks[J].arXiv: Computer Vision and Pattern Recognition, 2017.

[18]Lan G, Zhou Y. Conditional gradient sliding for convex optimization[J].Siam Journal on Optimization, 2016, 26(2): 1379-1409.

[19]毛先胤,刘宇,马晓红,等. 基于SSD算法的电力巡线机器人障碍物识别[J].自动化与仪器仪表,2020(5):45-48.

[20]张琳,马宏忠,王涛云,等. 基于振动-SVM的变压器绕组缺陷诊断方法[J].陕西电力,2016(11): 14-18.

[21]李川,李红英,谢旗辉,等. 基于可靠性大数据的电力公司经营决策模型设计[J]. 陕西电力,2016(10):62-66.