

目标跟踪中基于深度可分离卷积的剪枝方法

毛远宏, 贺占庄, 刘露露
(西安微电子技术研究所, 710065, 西安)

摘要: 为了减少跟踪网络中存在的参数量和计算量大的问题,提出了基于深度可分离卷积的剪枝方法。深度可分离卷积将跟踪网络中的传统卷积层分解为逐点卷积和逐层卷积两部分。在逐点卷积中,通过逐点卷积层中权重的大小来评估输入特征图通道在线性组合中的重要程度,将较小的权重及其关联的特征通道裁减掉。在逐层卷积中,通过 K-L 散度来衡量逐层卷积中滤波器的相似性,将相似的滤波器裁剪掉,减少冗余。通过上述方法进行多轮迭代剪枝,从而减少跟踪网络的参数量和计算量。在 VOT 数据集上的实验结果表明,在精度没有下降的前提下,剪枝后网络的参数量下降了 22.54%,计算量下降了 17.8%。在 NVIDIA TX2 设备上的实验结果表明,剪枝后网络的跟踪速度在 CPU 上提升了 14.95%,在 GPU 上提升了 13.07%。

关键词: 目标跟踪;卷积神经网络;深度可分离卷积;网络剪枝

中图分类号: TP391 **文献标志码:** A

DOI: 10.7652/xjtuxb202101007 **文章编号:** 0253-987X(2021)01-0052-08



OSID 码

Pruning Based on Separable Convolutions for Object Tracking

MAO Yuanhong, HE Zhanzhuang, LIU Lulu
(Xi'an Microelectronics Technology Institute, Xi'an 710065, China)

Abstract: A pruning method based on separable convolution is proposed to reduce parameters and calculation amount in the tracking network. Separable convolution decomposes the traditional convolutional layer into two parts, i. e. pointwise convolution and depthwise convolution in the tracking network. In pointwise convolution, the importance of the input feature map channel in the linear combination is evaluated by the weights in pointwise convolutions, so that the smaller weights and their associated feature channel ought to be pruned. In depthwise convolutions, the K-L divergence is adopted to measure the similarity of filters, and the similar filters are then pruned to reduce redundancy. With the improvements above, iterative pruning is performed. The parameters in the tracker are reduced by 22.54%, and the computation is reduced by 17.8%. On the NVIDIA TX2, the tracking rate increases by 14.95% for CPU and 13.07% for GPU without significant degradation of tracking performance in VOT benchmark.

Keywords: object tracking; convolutional neural networks; separable convolution; filter pruning

目前,基于深度学习的方法在目标跟踪领域得到了越来越多的应用,取得了很好的效果。但是,深

度学习通常存在着参数过多和计算量过大的问题。目标跟踪任务在保证精度的前提下,通常也需要考

收稿日期: 2020-06-06。 作者简介: 毛远宏(1980—),男,博士生;贺占庄(通信作者),男,研究员,博士生导师。 基金项目: 国家自然科学基金资助项目(61702413)。

网络出版时间: 2020-09-21

网络出版地址: <http://kns.cnki.net/kcms/detail/61.1069.T.20200920.1254.002.html>

考虑到实时性要求。因此,如何让目标跟踪任务的深度模型在存储和计算资源有限的设备更快地运行,是一个迫切需要解决的问题。

目标跟踪中的神经网络模型主要是通过卷积网络来提取相应的特征。绝大部分的计算量和参数主要集中在卷积层。因此,如何减少卷积层的计算,是加快跟踪速度的重点。除了提高硬件的计算能力之外,还可以通过优化网络模型,减少跟踪网络的参数量和计算量,加快跟踪速度。

如何对于神经网络进行加速是目前业界研究的热点。在卷积层的加速上,采用更加轻量化的卷积结构替换传统的卷积结构是通常采用的办法。例如:SqueezeNet 的 Fire 模块结构^[1],GoogLeNet 的 Inception 模块结构^[2],MobileNet 中的深度可分离卷积^[3]等。深度可分离卷积是在目前轻量化网络中被广泛使用的一种卷积结构。它可以在不改变网络总体层次架构的情况下,将传统的卷积层依次替换为深度可分离卷积。通过在训练集上进行重新训练,通常能够有效地降低运算量和参数量。基于深度可分离卷积的跟踪网络,参数量和计算量已经进行了很大精简。在此基础上,如何采用网络剪枝的方法进一步加快运行速度,也更加具有挑战性。

剪枝作为网络压缩的一种重要方法而被广泛使用^[4-5]。为了实验方便,之前大部分剪枝研究使用分类任务来评估方法效果,而在跟踪网络中的剪枝研究则很少涉及。因此,在实时性的要求下,跟踪网络剪枝非常有意义。网络剪枝按照裁剪的粒度,可以分为非结构化剪枝和结构化剪枝两种。非结构化剪枝方法^[6-7]主要通过网络稀疏化来实现,将不重要的权重设置为 0。但是,在卷积计算过程中,神经网络需要大量条件和额外空间来判断和表示非零参数及其位置,依赖于专门的运行库和硬件设备来进行加速。结构化剪枝方法^[8-11]通过删除滤波器或者特征图等结构化信息,减少网络的计算。裁减后的网络无需专门的软硬件来进行支持,使得剪枝方法有很好的通用性。

在跟踪任务中,可以采用结构化剪枝的方法对跟踪网络进行进一步精简。在剪枝过程中,本文根据深度可分离卷积的特点,充分利用深度分离卷积模型中的先验信息,更加有针对性地进行剪枝操作。

深度分离卷积包含逐层卷积和逐点卷积两个部分,其中逐点卷积占据了大部分的计算。剪枝的过程中,1×1 卷积被认为是对于输入特征图中不同通道的线性组合。在线性组合中,通常权重越大的特

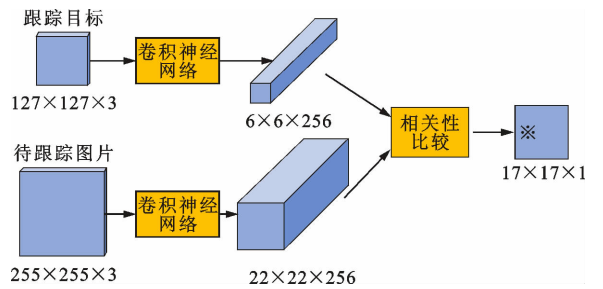
征图重要性越大,反之则重要性越小。在剪枝过程中,通过评估逐点卷积层中的权重大小,可以估计出输入通道的重要程度,从而将较小的权重及其关联的特征图裁减掉,降低网络的参数量和运算量。

逐层卷积中主要提取二维平面特征,不同的滤波器在空间平面上应该获取不同的特征信息。在深度可分离卷积中,逐层卷积的滤波器采用了单层的二维卷积结构,因此在滤波器相似性评估上更加简单。相似的滤波器获取的特征有相似之处,它们之间应该有相互替代的作用。通过 K-L 散度对于滤波器的相似性进行评估,删除相似的滤波器,从而减少卷积的计算量。

通过上述网络剪枝的方法对于以深度可分离卷积为主体的跟踪网络进行了精简。相比于其他剪枝方法,本文采用的剪枝方法在保证精度的前提下,能够更加有效地减少计算量,并在嵌入式设备 NVIDIA TX2 上取得较好的加速效果。

1 基于深度可分离卷积的 SiameseFC 网络

SiameseFC 网络^[12]是一种典型使用深度学习的跟踪网络,其通过两个卷积分支分别对于目标区域和整个图像进行特征提取,通过比较输出特征图的相关性来确定目标位置。SiameseFC 网络的思想对于基于深度学习的目标跟踪方法产生过重要影响,并在此基础上衍生出众多的基于深度学习的网络,例如 DASiamRPN^[13],SiamRPN^[14],SA-Siam^[15]等。SiameseFC 跟踪网络的基本框架结构如图 1 所示。



※—网络输出中目标所在的区域。

图 1 SiameseFC 跟踪网络架构

在图 1 中,使用 CNN 代表了跟踪任务中的卷积分支。在 SiameFC 层中使用 AlexNet^[16]作为基本网络,在训练的过程中共享了两个网络分支的权重。

为了降低计算量和参数量,深度可分离卷积将

传统的卷积分解为逐层卷积和逐点卷积两个部分。图2为深度可分离卷积的示意。在逐层卷积中,输入的 M 个通道特征图和 M 个滤波器分别各自进行卷积,输出 M 个特征图。在逐层卷积中,主要关注空间平面特征的提取,卷积计算仅限各个通道内部发生,通道之间的信息并不进行融合。在逐点卷积中,逐层卷积输出的 M 个特征图和 $1 \times 1 \times M$ 滤波器通过卷积操作输出一个特征图。由于在逐点卷积中有 N 个 $1 \times 1 \times M$ 卷积核,因此最终输出 N 个通道的特征图。在这个过程中,通过逐点卷积实现了输入特征图在通道层面的特征融合。深度可分离卷积主要改变了传统卷积的内部实现,对外依然可以保持之前输入和输出的特征维度。因此,可以在不修改网络层级结构的情况下,将相应的传统卷积依次替换为深度可分离卷积。

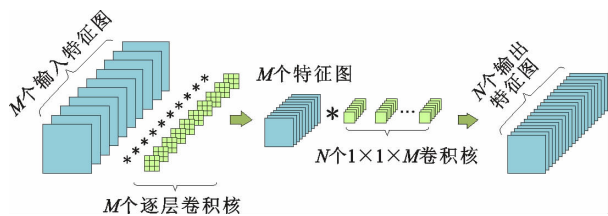


图2 深度可分离卷积结构示意图

在计算量上,传统卷积的计算量为 $D_K D_K M N D_F D_F$,其中, D_K 为卷积的大小, D_F 为输入特征图的大小。使用深度可分离卷积之后,传统的卷积被分解为逐层卷积和逐点卷积,其计算量为两者之和 $D_K D_K M D_F D_F + M N D_F D_F$,其中, $D_K D_K M D_F D_F$ 是逐层卷积的计算量, $M N D_F D_F$ 是逐点卷积的计算量。优化后与优化前的计算量之比为

$$\frac{D_K D_K M D_F D_F + M N D_F D_F}{D_K D_K M N D_F D_F} = \frac{1}{N} + \frac{1}{D_K^2} \quad (1)$$

以目前通常使用的 3×3 卷积核为例, D_K 为3, $1/D_K^2$ 为 $1/9$ 。 N 为输出通道数,一般比较大,因此 $1/N$ 通常可以忽略不计。从式(1)可以看出,优化后逐点卷积占据了绝大部分的计算量。

2 逐点卷积中基于权重的剪枝

在可分离卷积中,绝大部分的计算量还是集中在逐点卷积上。因此,首先将剪枝的重点放在逐点卷积上。

M 个输入特征图($F_1, F_2, \dots, F_M$)和一个 $1 \times 1 \times M$ 滤波器($k_1, k_2, \dots, k_M$)的卷积过程可描述为

$$\begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_M \end{bmatrix} * \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_M \end{bmatrix} = \sum_{i=1}^M F_i k_i \quad (2)$$

最终输出的特征图为特征图的叠加 $\sum_{i=1}^M F_i k_i$,其中 $F_i k_i$ 表示权重系数 k_i 和特征图 F_i 上每个元素相乘。由于逐点滤波器的平面尺度为 1×1 , k_i 为权重系数,所以逐点卷积的输出可以被认为输入特征图的多项线性组合。

逐点卷积计算可以表示为特征图的线性组合。在多个 $1 \times 1 \times M$ 卷积中,若某通道对应的权重很小,则可以认为这个通道的重要性不强。在剪枝过程中,将不重要的特征图连同其权重裁减掉,如图3所示,图中 $1 \times 1 \times M$ 滤波器中,红色输入特征图对应的权重很小。

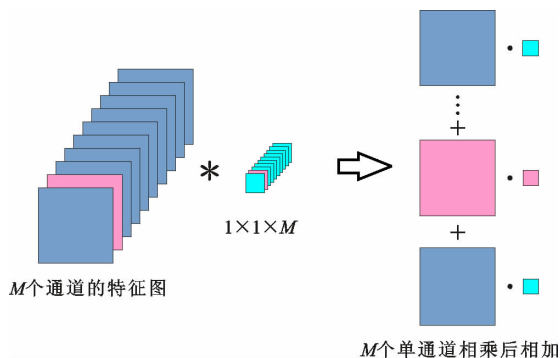


图3 对于权重小的输入特征图进行剪枝

在逐点卷积裁剪过程中,按照逐点卷积每层分别进行剪枝。先对于 1×1 滤波器权重进行归一化,再按照权重的大小对于特征图重要性进行排序。在剪枝过程中,以超参数 α 为比例,按照权重的大小进行剪枝。超参数 α 为通过实验获得的经验值。

3 逐层卷积中基于相似性的剪枝

在逐层卷积中,主要是针对图像的空间信息进行特征提取。在保持神经网络精度的前提下,功能相似的滤波器具有相互的补充作用。即使其中一个滤波器被删除,在重训练的过程中,剩余相似的滤波器依然可以保证网络精度。通过减少此种冗余,可以直接减少参与运算的滤波器,从而降低网络的复杂度。

采用两个滤波器之间的对称性 K-L 散度来评估其相似性,公式为

$$D_{KL}(P, Q) = \frac{1}{2} (D_{KL}(P \parallel Q) + D_{KL}(Q \parallel P)) =$$

$$\frac{1}{2} \left(\sum_{i=1}^N p(x_i) \log \frac{p(x_i)}{q(x_i)} + \sum_{i=1}^N q(x_i) \log \frac{q(x_i)}{p(x_i)} \right) \quad (3)$$

式中: P 和 Q 表示待评估两个滤波器的概率分布; x_i 表示滤波器的值; $p(x_i)$ 和 $q(x_i)$ 分别表示在滤波器相应位置上的权重占比。在对比过程中, 首先将滤波器的权重进行归一化, 然后将其权重按照所在位置进行排列, 对于两个滤波器的权重分布进行相似性比较。

为了保留信息更加丰富的滤波器, 通过交叉熵评估相似的滤波器。保留交叉熵较大的滤波器, 裁减交叉熵比较小的滤波器。

空间逐层卷积中基于相似性的剪枝方法流程如下。

输入: 逐层卷积的 M 个滤波器, 剪枝超参数 β

输出: 被移除的 $M\beta$ 个滤波器

步骤 1 对于当前层 M 个滤波器进行归一化预处理。在计算两个滤波器散度的过程中, 如果滤波器参数存在负值, 则先在滤波器中所有权重上添加最小值的绝对值。接着对于每个位置上的权重, 通过 $w_{ij} / \sum_{i=1}^s \sum_{j=1}^s w_{ij}$ 进行归一化, 其中, w_{ij} 表示在滤波器 (i, j) 位置上的权重, s 是滤波器的大小。

步骤 2 对于 M 个滤波器进行两两之间 K-L 散度计算, 并将按照 K-L 散度从小到大将滤波器进行排序。

步骤 3 以超参数 β 为比例, 选择 $M\beta$ 对相似的滤波器, 作为待剪枝的滤波器。

步骤 4 对于待剪枝的滤波器对, 分别计算两个滤波器的交叉熵, 裁减交叉熵小的滤波器。

步骤 5 逐层卷积剪枝完成后, 通过迭代训练恢复跟踪网络的精度。

在剪枝过程中以超参数 β 为比例进行剪枝处理, 此参数是通过实验得出的最优比例。

4 实验

4.1 数据集

在训练过程中, 与原始 SiameseFC 网络保持一致, 采用 ImageNet 视频训练集。此视频训练集包含 4 000 多段视频, 超过百万具有标注的视频帧。对于跟踪任务的训练, 它不仅包含了更多的视频片段, 并且在目标和场景上也超过其他训练集。因此, ImageNet 视频训练集非常适合用于训练基于深度学习的跟踪网络, 避免出现过拟合情况。

在测试过程中, 为了保证与原始 SiameseFC 跟踪网络相同的实验条件, 同样使用 VOT 数据集^[17]作为评估数据集。从 2013 年开始, VOT 数据集成为了一个重要的单目标跟踪测试平台。VOT 数据集不仅有大量的测试数据, 并且综合考察了遮挡、形变、模糊、分辨率、运动等因素对于跟踪任务中的影响。

4.2 测试方法

在深度分离卷积为主的训练过程中, 首先使用深度卷积网络替换传统网络, 然后重新开始训练。采用 Xavier^[18]方法来初始化网络参数。采用随机梯度下降方法来优化跟踪网络, minibatch 设置为 8, 执行 50 轮训练迭代, 学习率从 10^{-2} 到 10^{-5} 按照迭代次数均匀下降。通过训练, 得到了基于深度分离卷积的 SiameseFC 跟踪网络。

为了减少剪枝对于深度可分离卷积跟踪网络准确度的影响, 在剪枝的过程中, 采用逐次剪枝的方法, 即每次进行少量剪枝, 然后通过少量轮迭代训练, 使得网络精度得到恢复, 再进入下一轮剪枝。在网络剪枝中, 逐点卷积剪枝超参数 α 设置为 0.10, 逐层卷积剪枝超参数 β 设置为 0.05。将待裁剪的通道或者滤波器分散到 10 轮剪枝迭代中进行。每次剪枝后, 进行 2 到 3 轮训练来恢复网络精度。

所有的训练在 Nvidia TITAN XP GPU 硬件设备上进行。使用 Matlab 的 MatConvNet^[19]进行训练。在测试过程中, 由于 TensorFlow 平台对于深度可分离卷积实现上支持力度更好, 因此将训练好的权重导入到 TensorFlow 环境中进行前向推理测试。

4.3 裁剪可视化

在逐点卷积的裁剪过程中发现, 权重较大的特征图通常包含比较多的信息, 相反, 权重较小的特征图没有明显的特征。图 4 是 SiameseFC 网络中第 3 层卷积的输入特征图可视化结果。在图 4b 中, 输入特征图按照其在逐点卷积中的权重大小进行从上到下、从左到右排序, 上半部分为 1×1 卷积中大权重所对应的特征图, 下半部分为较小权重的特征图。可以看出: 权重较大的特征图一般特征信息都比较丰富, 权重较小的特征图信息量通常比较缺乏。

图 5 为逐层卷积的裁剪过程中, 跟踪网络中第 2 个逐层卷积内 K-L 散度 top5 最小的滤波器对, 图中一列为一组滤波器对。可以看出, 这些滤波器对之间有明显的相似性。

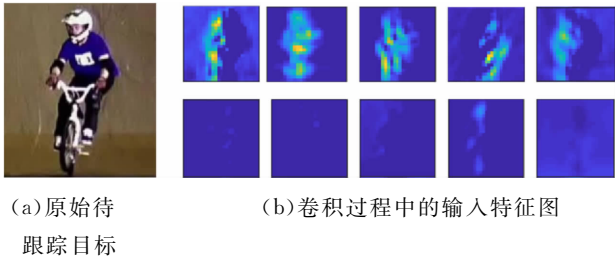


图 4 逐点卷积中的输入特征图可视化

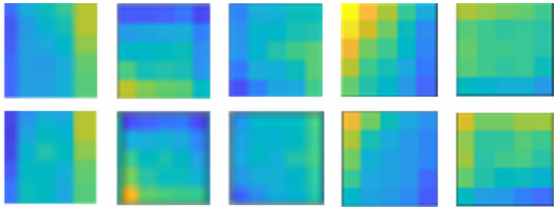


图 5 逐层卷积中相似的滤波器组可视化

4.4 测试结果

为了验证本文的剪枝方法,在不同的网络架构上进行了相应的对比实验,比较了计算量和精度。进行对比的网络架构如下。

表 1 不同 SiameseFC 网络的性能比较

SiameseFC 网络类型	准确率	失败率	平均重叠期望	参数量 /10 ⁶	乘加计算量 /10 ⁶
传统卷积	0.54	2.05	0.247 2	2.334	3 318.13
深度可分离卷积	0.53	1.76	0.243 5	0.417	935.91
基于深度可分离卷积的随机剪枝 ^[20]	0.48	3.49	0.144 6	0.350	817.82
基于深度可分离卷积的权重绝对值剪枝 ^[8]	0.54	2.37	0.225 9	0.326	790.81
基于深度可分离卷积以 APoZ 为标准的剪枝 ^[11]	0.54	2.12	0.236 0	0.348	810.75
基于深度可分离卷积的逐层卷积方差剪枝 ^[21]	0.55	2.34	0.246 6	0.317	790.06
本文方法剪枝	0.55	2.32	0.247 0	0.323	768.61

在表 1 中,跟踪目标的准确率通过视频中预测框和真实区域的交并比平均值进行评估。交并比越大,准确率越高。失败率是每段视频中的平均失败帧数,用来衡量跟踪的鲁棒性。当交并比为 0 时,认为跟踪失败,跟踪网络将重新进行初始化。平均重叠期望(EAO)是考虑到准确性和鲁棒性的综合指标。EAO 用来衡量在跟踪网络未发生失败情况下的平均准确率,是衡量跟踪网络性能最重要的指标。参数量是指 CNN 分支的参数量。由于两个 CNN 分支共享参数,因此在表 1 中的参数量是指单个网络的参数量。计算量是两个分支的卷积中乘加计算的量。由于两个分支输入的图像大小不同,因此

- (1)传统卷积 SiameseFC 网络。
 - (2)深度可分离卷积 SiameseFC 网络。Siamese FC 网络的传统卷积被深度可分离卷积替代,剪枝操作以此作为基准。
 - (3)随机剪枝的深度可分离卷积 SiameseFC 网络^[20]。
 - (4)在深度可分离卷积 SiameseFC 网络上,按照滤波器的权重绝对值大小(L1 范数)进行剪枝^[8]。将卷积层中权重绝对值大的滤波器保留下来,绝对值小的滤波器裁剪掉。
 - (5)在深度可分离卷积 SiameseFC 网络上,将激活层输出为 0 元素的平均百分比(APoZ)^[11]作为评价滤波器是否重要的标准,进行剪枝。
 - (6)在深度可分离卷积 SiameseFC 网络的逐层卷积中,按照滤波器的方差大小逐层进行剪枝^[21]。滤波器的方差越小,其作用越小,应该优先被裁剪掉。
- 本文方法和 6 种对比方法的性能比较如表 1 所示。

两个分支的计算量不同。表 1 中的计算量为两个分支的卷积计算量总和。

从表 1 可知,本文剪枝方法在跟踪精度和计算效率方面取得了良好的效果。相比于剪枝之前的深度可分离卷积 SiameseFC 网络,在网络性能没有下降的情况下,网络的参数量下降了 22.54%,卷积层的计算量下降了 17.8%。在保持精度的同时,将运算量降到了最少。

为了进一步证明本文剪枝方法的有效性,采用了 NVIDIA TX2 进行了进一步的实验。NVIDIA TX2 是嵌入式移动设备中普遍使用的一款硬件。在 Tensorflow 平台上进行了测试验证,结果如表 2 所示。

表 2 跟踪网络在 NVIDIA TX2 设备上的计算速度对比

SiameseFC 网络类型	平均重叠 期望	参数量 /10 ⁶	乘加计算量 /10 ⁶	CPU 计算速度 /(帧·s ⁻¹)	GPU 计算速度 /(帧·s ⁻¹)
传统卷积	0.247 2	2.334	3 318.13	0.98	6.59
深度可分离卷积	0.243 5	0.417	935.91	3.21	12.40
本文方法	0.247 0	0.323	768.61	3.69	14.02

从表 2 可以看出,在嵌入式设备上,相比于深度可分离卷积 SiameseFC 网络,本文方法剪枝后网络的计算速度在 CPU 上提升了 14.95%,在 GPU 上提升了 13.07%,在 CPU 和 GPU 的速度提升比例不完全相同。这是因为在 GPU 中,内存数据的访问频率和速度对于计算速度也会有较大的影响。

为了充分说明采用本文方法剪枝后网络的泛化能力,采用本文剪枝方法精简了网络,在 OTB50 数据集^[22]上进行了测试验证。图 6 是 SiameseFC 网络剪枝前后在 OTB50 数据集上的指标对比。可以看出:当位置误差阈值为 20 时,本文方法、传统卷积、深度可分离卷积 SiameseFC 网络的准确率分别

为 0.692、0.682、0.613;当重叠阈值为 0.5 时,本文方法、传统卷积、深度可分离卷积 SiameseFC 网络的成功率分别为 0.600、0.621、0.561;剪枝后网络的各指标均超过剪枝前的。

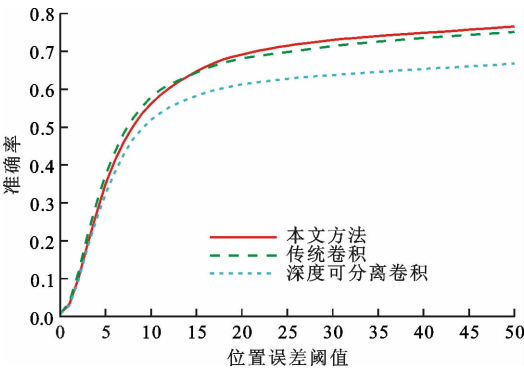
通过对比剪枝前后的测试结果可以看出,剪枝后的网络在 VOT 数据集的 EAO 指标,以及 OTB50 数据集上的准确率和成功率指标都有所提升。从机器学习的理论分析,网络的泛化能力在一定程度上和网络自身的复杂度成反相关^[23]。因此,适当的剪枝减少了参数量和计算量,降低了模型的复杂度,提升了跟踪网络的泛化能力。剪枝后网络在数据集上的精度超过了剪枝前网络的。

图 7 展示了本文方法剪枝后的网络在 VOT 数据集上的结果。可以看出:在前 3 行跟踪序列中,目标存在较大遮挡,剪枝后的跟踪网络依然能够保持很好的跟踪效果;第 4 行跟踪序列为有很大运动模糊的场景,第 5 行跟踪序列为光照变化剧烈的场景,第 6 行跟踪序列为跟踪物体尺度变化的场景,在这些场景中,剪枝后的网络都可以保持优异的性能。

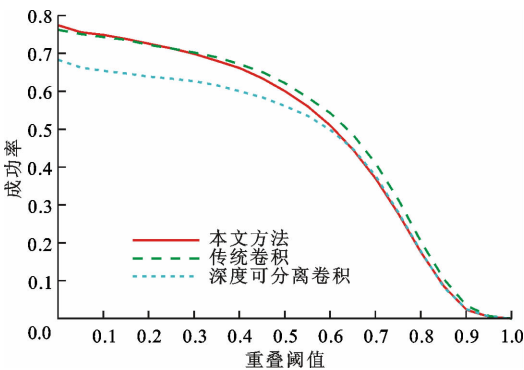
5 结 论

随着深度学习在跟踪领域的广泛应用,众多的参数量和计算量对于跟踪实时性是一个巨大的挑战。深度可分离卷积相比传统卷积,计算量已经有较大精简,在此基础上进行剪枝更加具有挑战性。

本文根据深度可分离卷积的特点,有针对性地对网络进行剪枝,进行进一步加速。剪枝过程中,通过在逐点卷积上评估输入特征图通道的线性系数大小,将不重要的通道裁减掉。在逐层卷积中,衡量空间卷积滤波器的相似性,减少滤波器的冗余。在 VOT 数据集上的结果表明,相比未剪枝的网络,本文方法剪枝后的网络能够在精度没有下降的情况下,使网络参数量下降 22.54%,计算量下降 17.8%。在嵌入式设备 NVIDIA TX2 上,本文方法剪枝后的网络在 CPU 上的计算速度提升了 14.95%,在 GPU 上提升了 13.07%。



(a)准确率



(b)成功率

图 6 SiameseFC 网络剪枝前后在 OTB50 数据集上的指标对比

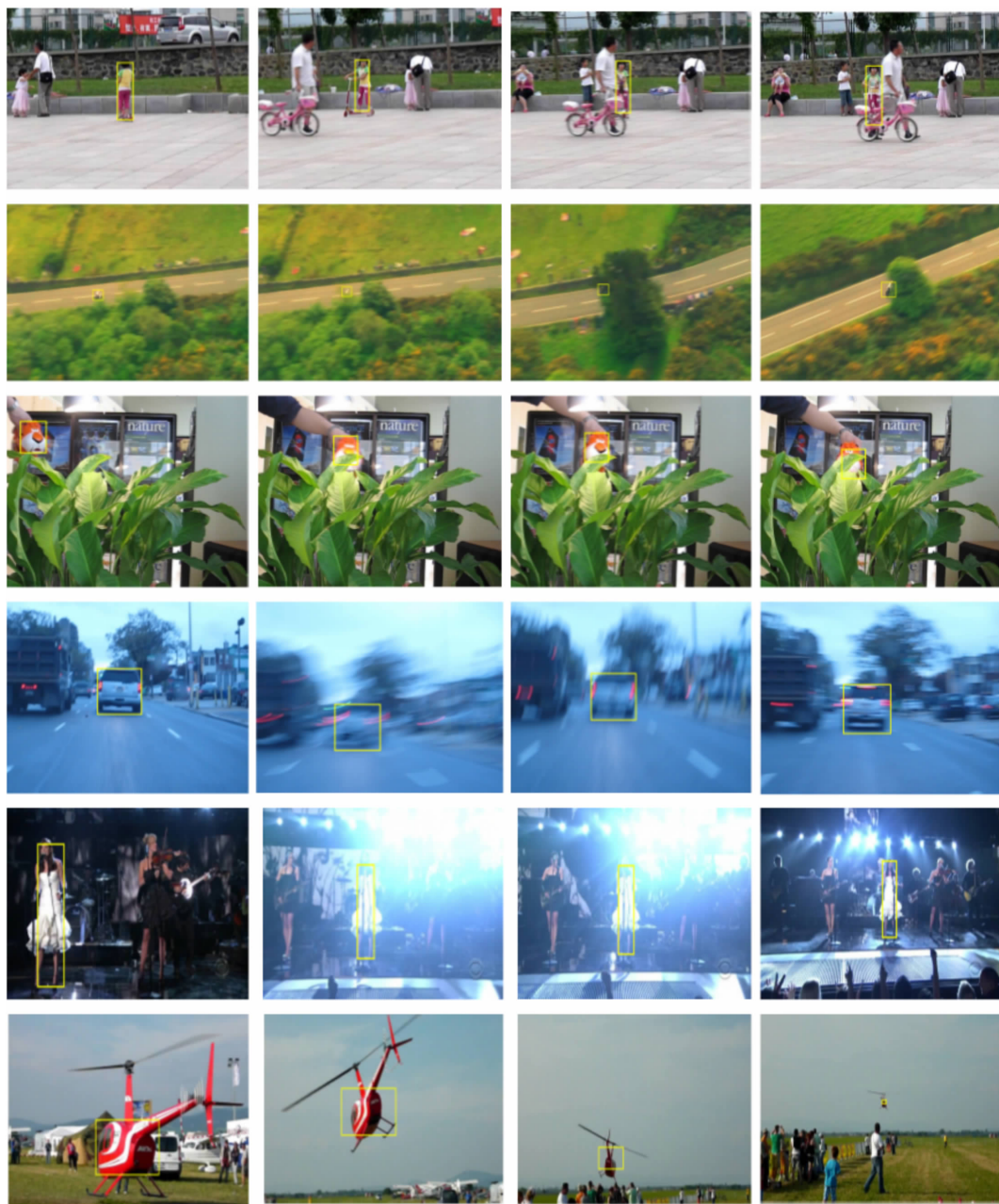


图 7 本文方法剪枝后网络在 VOT 数据集上的跟踪效果

参考文献:

- [1] IANDOLA F N, HAN SONG, MOSKEWICZ M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size [EB/OL]. [2020-04-01]. <https://arxiv.org/abs/1602.07360>.
- [2] CHOLLET F. Xception: deep learning with depthwise separable convolutions [C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2017: 1800-1807.
- [3] HOWARD A, ZHU Menglong, CHEN Bo, et al. MobileNets: efficient convolutional neural networks for mobile vision applications [EO/OL]. [2020-04-01]. <https://arxiv.org/abs/1704.04861>.
- [4] 孙彦丽, 叶炯耀. 基于剪枝与量化的卷积神经网络压缩方法 [J]. 计算机科学, 2020, 47(8): 261-266.
- [5] SUN Yanli, YE Jiongyao. Convolutional neural networks compression based on pruning and quantization [J]. Computer Science, 2020, 47(8): 261-266.
- [6] 林景栋, 吴欣怡, 柴毅, 等. 卷积神经网络结构优化综述 [J]. 自动化学报, 2020, 46(1): 24-37.
- [7] LIN Jingdong, WU Xinyi, CHAI Yi, et al. Structure optimization of convolutional neural networks: a survey [J]. Acta Automatica Sinica, 2020, 46(1): 24-37.

- [6] WEN Wei, WU Chunpeng, WANG Yandan, et al. Learning structured sparsity in deep neural networks [EB/OL]. [2020-04-01]. <https://arxiv.org/abs/1608.03665>.
- [7] HAN Song, POOL J, TRAN J, et al. Learning both weights and connections for efficient neural networks [EB/OL]. [2020-04-01]. <https://arxiv.org/abs/1506.02626>.
- [8] LI Hao, KADAV A, DURDANOVIC I, et al. Pruning filters for efficient convNets [C]// Proceedings of the 2017 International Conference on Learning Representations. London, UK; ICLR, 2017:1-12.
- [9] LUO Jiahao, WU Jianxin. An entropy-based pruning method for CNN compression [EB/OL]. [2020-04-01]. <https://arxiv.org/abs/1706.05791>.
- [10] LIU Zhuang, LI Jianguo, SHEN Zhiqiang, et al. Learning efficient convolutional networks through network slimming [C]// Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV). Piscataway, NJ, USA; IEEE, 2017: 2755-2763.
- [11] HU Hengyuan, PENG Rui, TAI Yuming, et al. Network trimming: a data-driven neuron pruning approach towards efficient deep architectures [EB/OL]. [2020-04-01]. <https://arxiv.org/abs/1607.03250>.
- [12] BERTINETTO L, VALMADRE J, HENRIQUES J F, et al. Fully-convolutional Siamese networks for object tracking [C]// Proceedings of the 2016 European Conference on Computer Vision. Cham, Germany; Springer, 2016: 850-865.
- [13] LI Bo, YAN Junjie, WU Wei, et al. High performance visual tracking with Siamese region proposal network [C]// Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA; IEEE, 2018: 8971-8980.
- [14] ZHU Zheng, WANG Qiang, LI Bo, et al. Distractor-aware Siamese networks for visual object tracking [C]// Proceedings of the 2018 European Conference on Computer Vision. Cham, Germany; Springer, 2018: 103-119.
- [15] HE Anfeng, LUO Chong, TIAN Xinmei, et al. A twofold Siamese network for real-time object tracking [C]// Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA; IEEE, 2018: 4834-4843.
- [16] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks [J]. Communications of the ACM, 2017, 60 (6): 84-90.
- [17] KRISTAN M, MATAS J, LEONARDIS A, et al. The visual object tracking VOT2015 challenge results [C]// Proceedings of the 2015 IEEE International Conference on Computer Vision Workshop (ICCVW). Piscataway, NJ, USA; IEEE, 2015: 564-586.
- [18] HE Kaiming, ZHANG Xiangyu, REN Shaoqing, et al. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification [C]// Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV). Piscataway, NJ, USA; IEEE, 2015: 1026-1034.
- [19] VEDALDI A, LENC K. MatConvNet: convolutional neural networks for MATLAB [C]// Proceedings of the 23rd ACM International Conference on Multimedia. New York, USA; ACM, 2015: 689-692.
- [20] MITTAL D, BHARDWAJ S, KHAPRA M M, et al. Studying the plasticity in deep convolutional neural networks using random pruning [J]. Machine Vision and Applications, 2019, 30(2): 203-216.
- [21] MAO Yuanhong, HE Zhazhuang, MA Zhong, et al. Efficient convolution neural networks for object tracking using separable convolution and filter pruning [J]. IEEE Access, 2019, 7: 106466-106474.
- [22] WU Y, LIM J, YANG M H. Online object tracking: a benchmark [C]// Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA; IEEE, 2013: 2411-2418.
- [23] VAPNIK V, LEVIN E, CUN Y L. Measuring the VC-dimension of a learning machine [J]. Neural Computation, 1994, 6(5): 851-876.

(编辑 陶晴)