基于模块相似性的超分网络剪枝

周仁爽[1,2]，陈尧森[1,2]，郭 兵[1*]，沈 艳[3]，李 杰[2]，王 炜[1,2,4]

(1. 四川大学计算机学院 成都 610065；2. 成都索贝数码科技股份有限公司 成都 610041；
3. 成都信息工程大学计算机学院 成都 610225；4. 鹏城实验室 广东 深圳 518055)

【摘要】该文针对单图像超分辨率网络 (SISR) 提出了一种简单的网络剪枝方法。该方法通过评估超分网络中各模块的相似性，用一种简单办法将相似度转换为各模块对网络的贡献程度，从而找到对超分网络相对不重要的模块进行网络剪枝，达到超分辨率网络压缩的目的。通过基于模块相似性的超分网络剪枝，原本参数量庞大的超分网络得到了压缩，参数量和运算量都大幅下降。实验表明，通过剪枝后的超分网络其参数量可以下降 60% 以上，同时精度下降不超过 0.1%，对超分网络部署到低性能平台有着实际意义。

关 键 词 超分辨率； 网络压缩； 模块相似性； 网络剪枝
中图分类号 TP183 **文献标志码** A doi:10.12178/1001-0548.2021126

Module Similarity-Based Pruning for Image Super-Resolution Network

ZHOU Renshuang[1,2], CHEN Yaosen[1,2], GUO Bing[1*], SHEN Yan[3], LI Jie[2], and WANG Wei[1,2,4]

(1. College of Computer Science, Sichuan University Chengdu 610065; 2. Chengdu Sobey Digital Technology Co., Ltd Chengdu 610041;
3. School of Computer Science, Chengdu University of Information Technology Chengdu 610225;
4. Peng Cheng Laboratory Shenzhen Guangdong 518055)

Abstract This paper proposes a network pruning method for single image super-resolution network (SISR). This method evaluates the similarity of each module in the super-resolution network and uses a simple method to convert the similarity into the contribution degree of each module to the network, and find the relatively unimportant modules of the network to perform network pruning. Through the method of network pruning for the super-resolution network based on the module similarity, the super-resolution network with a huge amount of parameters is compressed, and the number of parameters and the amount of calculation are greatly reduced. Experiments show that the parameters of the super-resolution network after pruning can be reduced by more than 60%, while the accuracy is not reduced by more than 0.1%, which has great practical significance for the deployment of the super-resolution network to a low-performance platform.

Key words image super-resolution network; model compression; module similarity; network pruning

单图像超分 (single image super resolution, SISR) 是一种经典的机器视觉任务，其目的是从低分辨率图像中重构出高分辨率图像。图像超分被广泛应用于许多机器视觉的任务中，如医学影像[1]、监控影像[2]、目标识别[3] 等，其巨大的应用前景，成为了机器视觉领域的研究热点。

随着卷积神经网络 (convolution neural network, CNN)[4] 的出现，基于深度学习的超分网络因其强大的特征表达能力在图像超分领域取得了优异的表现，并逐渐在图像超分领域中占据了主导地位。文献 [5] 最早提出了基于 CNN 的图像超分网络 (super-resolution convolutional neural network, SRCNN)，其从稀疏编码[6-7] 中汲取灵感，使用了一个 3 层的 CNN 结构实现了低分辨率到高分辨率的图像重建。此后基于 CNN 的超分网络被不断提出，并一直刷新着超分网络的最佳性能表现。其中 SRResNet[8]、EDSR[9] 等网络采用了类似文献 [10] 提出的残差网络结构，构建出了由残差模块堆砌出的具有相当深

收稿日期：2021－05－06；修回日期：2021－08－17
作者简介：周仁爽 (1996－)，男，主要从事深度神经网络压缩方面的研究.
*通信作者：郭兵，E-mail：guobing@scu.edu.cn

度的网络结构。文献 [11] 更是提出了 RIR(residual in residual) 的结构，将网络深度提高到了 400 多层，取得了惊人的性能表现。然而在超分网络深度不断加深的同时，基于 CNN 的超分网络也面临着资源消耗越来越大的难题。在实际应用中，更深的超分网络带来了出色的性能表现，但同时也带来了庞大的参数量和浮点操作计算量 (FLOPs)，如 RCAN (residual channel attention networks) 便拥有着 30×10^9 的 FLOPs 以及 13×10^6 的参数量 (Params)。如此庞大的计算量和内存消耗对于一些性能和存储有限的平台，特别是对移动平台和嵌入式平台的移植工作提出了巨大的挑战。因此对复杂的超分模型进行压缩优化，是非常有必要的。

网络模型压缩的目的在于尽可能地降低模型参数量和计算量，同时又不能出现明显的精度下降。目前常见的网络压缩方法有量化 (quantization)[12-14]、知识蒸馏 (konwledge distillation, KD)[15-17] 和网络剪枝 (network pruning)[18-22]。量化是一种像素级别的压缩方法，通过将全精度 (32 bit) 的权重 (weights)、激活值 (activations) 以及梯度值 (gradients) 量化到低精度 (如 8 bit)，从而达到压缩和加速网络的目的。然而量化的方法需要软硬件都支持低精度运算，在使用范围上大幅受限，并且容易带来模型精度的明显下降，并不适合所有网络。而知识蒸馏则是使用一个复杂强大的教师网络来监督简单小巧的学生网络训练，并将教师网络学到的知识提炼给学生网络，在模型压缩上有较好的效果。但知识蒸馏的方法需要合理地设计教师网络和学生网络，在实际使用中缺乏灵活性。相反，网络剪枝在模型压缩的方法中具备较高的灵活性，对大部分网络都能适用，且能直接有效地减少模型参数，降低模型的存储消耗，并加速模型推理，在模型压缩领域有着广泛应用。

网络剪枝作为一种常见的模型压缩方法，在一些高级别的机器视觉任务中已经得到了广泛应用，并证明了其有效性。但是在基于 CNN 的超分网络中却鲜有使用，因为如 EDSR、RCAN 等超分网络都具有独特的网络结构，如果采用常见的通道剪枝或权重稀疏等网络剪枝方法，可能会破坏原有的网络结构，造成较大的精度损失。

为了解决这个问题，本文提出了一种模块重要性的评估方法，用于评估 EDSR 等网络中每个残差模块对于网络的贡献程度，并移除对网络贡献度较小的模块。由于是对模块整体进行删减，因此并没有破坏网络的特殊结构，在剪去大量参数的同时也最大限度地保留了模型原有的精度。

本文的主要贡献有两点：1) 提出了一种评估超分残差模块重要性的方法，该方法具有通用性，可以用于大部分超分网络。2) 提出了一种超分网络剪枝的方法，通过网络剪枝降低超分网络的参数量以及运算量，降低网络的部署难度。

1　相关工作

1.1　基于深度学习的超分模型

自 SRCNN[5] 首先将深度学习的方法用到图像超分任务上，大量基于深度学习的超分网络被相继提出。VDSR[23] 通过一个残差结构解决了网络加深所产生的梯度爆炸问题，同时通过堆积卷积核的方式获得一个较大的感受野，解决了 SRCNN 受限于小感受野的问题。VDSR 使用了一个深的神经网络模型对低分辨率图像进行重构，并将残差结构引入超分网络，对此后很多超分网络的设计产生了影响。文献 [23] 认为更深的网络能够提供更大的感受野，帮助超分网络更好的重构画面细节。在这种思想的指导下，诞生了不少深度颇深的网络结构，以 EDSR[9]、RCAN[11]、RDN[24] 为代表的网络通过堆叠残差模块 (resblock)的方式解决了网络加深时带来的训练困难问题，并取得了 state-of-the-art 的成绩。

然而因为网络深度的加深，计算开销也随之而来，这使得将网络移植到一些硬件资源有限的设备上非常困难。为了降低网络的复杂度，使之可以部署到低性能平台，网络剪枝是一种值得考虑的方式。

1.2　网络剪枝

文献 [25-27] 通过重新设计高效网络来实现降低模型参数量的目的。而网络剪枝则是从一个大网络通过压缩的方式来获得一个更加高效的小网络，泛用性更高，避免了设计网络的高门槛。神经网络模型通常是过参数的[28]，包括了很多冗余参数，而网络剪枝的目的就是移除这部分对网络来说不重要的参数。从网络剪枝作用的层级上来说，剪枝分为非结构化剪枝和结构化剪枝。

早期的剪枝工作大都集中在非结构化剪枝，非结构化剪枝也即权重剪枝，这种剪枝方法直接作用于单个神经元的权重，可以最大化地移除冗余连接，实现最佳剪枝率。文献 [29-30] 通过移除网络中的绝对值较小的权重，将 AlexNet[31] 的参数量降低了 9 倍，而 VGGNet[32] 更是将参数量降低了 13 倍，

从 138 M 降低至 10.3 M，取得了优秀的压缩效果。然而虽然非结构化剪枝的效果十分强大，但是由于对每个神经元都剪去了不同数目的连接，导致每个节点输入和输出数目不规则。这种稀疏的结构无法利用现有的 BLAS 库加速矩阵运算，因此即便模型的参数降低了，模型的推理速度却没有实质性的提升。

结构化剪枝通常裁剪的是网络结构的某部分，如通道剪枝、层剪枝，而不是单独的某个权重。剪枝后的网络结构不会变得稀疏，因此结构化剪枝并不需要依赖特殊的软件库 (如稀疏矩阵运算) 支持，便可以直接实现模型的推理加速，相比于非结构化剪枝更加具有优势。文献 [33] 将网络中每个卷积核矩阵按绝对值进行求和，将得到较小值的通道从网络中移除，实现了 VGGNet 推理成本下降 34%，ResNet110 推理成本下降 38% 的加速效果。文献 [34] 则是利用 Batch Normalization 中的缩放因子的大小来定义对应通道的重要性，并且为了约束 BN 层中缩放因子的大小，在目标方程中添加了一个稀疏正则项，使得更多缩放因子在训练中接近于 0，以此提高剪枝率。以上的通道剪枝方法在图像分类等高级机器视觉任务中取得了不错的压缩效果，然而在图像超分等低级任务中却实践较少。原因是目前基于 CNN 的图像超分网络平等地看待每个通道的特征，如果移除了部分通道可能会带来无法接受的精度损失。为了避免通道剪枝影响超分网络的特殊结构，造成较大的精度损失，本文将剪枝的范围落到了超分网络的模块 (block) 上。由于大部分超分网络都是由相同的模块堆砌而成，只减少模块数量并不会破坏网络的原本结构，因此模块剪枝或者说层剪枝是更适合超分网络的剪枝方法。

1.3 层剪枝

层剪枝也是结构化剪枝的一类，相比于通道剪枝，层剪枝将剪枝的范围扩大到层级别，剪枝范围更大，能减少的参数数量也更多。文献 [35] 利用了文献 [36] 中提出的线性探针技术，计算出 CNN 网络中的每一层对于网络整体的贡献程度，并通过移除对网络贡献度较低的层来达到网络剪枝的目的。结合知识蒸馏后，能做到在精度几乎无损失甚至是略好于原模型的情况下大幅削减模型参数量。文献 [37] 中裁剪了图像超分网络 RCAN 和 SAN[38] 的模块数量，并利用知识蒸馏 (KD) 的方法来恢复模型精度，最终在精度下降不多的情况下取得了较好的模型压缩效果。此方法有效地压缩了超分网络的大小，实现了性能提升，但是在选择要剪枝的模块时并没有任何指导性，仅仅是减少模块数量再借由知识蒸馏恢复网络精度。

为了解决这个问题，本文提出了一种评估模块重要性的方法，针对性地移除重要性较低的模块。由于被剪模块的选择更具有指导性，剪枝后的网络仅需要简单的微调 (fine-tune) 即可恢复到比较理想的精度，避免了知识蒸馏的高额时间成本，实现超分网络压缩的目的。

2 基于模块相似性的网络剪枝

本文方法是一种针对于超分网络模块的网络剪枝方法，其主要目的是移除对网络贡献不大或不重要的模块，以达到网络轻量化的目的。因此也属于层剪枝的范畴，为了避免混淆，以下统称为模块剪枝。

2.1 模块重要性评估方法

常见的 CNN 超分网络通常可分为特征提取和图像重构两部分。VDSR 提出更深的网络结构能够获得更大的感受野，从而在图像重构时得到的超分图像获得更高的质量。随着残差网络结构的引入，增加了超分网络的深度，提高了网络的超分性能。由残差模块 (Resblock) 堆砌而成的超分网络结构，如图 1 所示，成为了一种比较主流的超分网络结构。

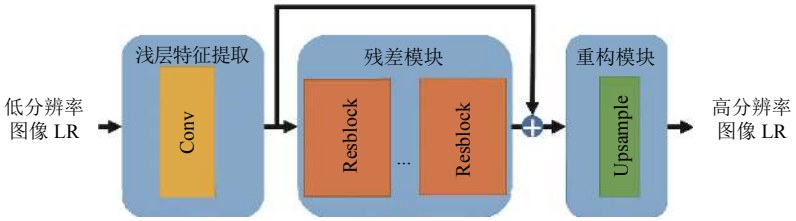


图 1 超分网络常见结构

基于残差模块堆砌的超分网络并不是直接预测超分图像，而是预测插值算法得到的插值图像相比于超分图像的残差。因此只要能够保证网络预测得到的残差是正确的，就能最大程度保证网络的精度。因为最后一个残差模块的输出即为网络预测的残差，在此前提下，本文以网络中最后一个残差模

块的输出作为标准输出，并以余弦相似度作为相似度指标，计算网络中其他模块的输出和这个标准输出的相似度。如图 2 所示，以 EDSR 为例，计算的各残差模块相对于最后一个残差模块输出的相似度，图中横坐标代表每个残差模块的标号，纵坐标代表该模块相对于最后一个模块的余弦相似度。可以看到越靠后的模块，输出相对于标准输出就越接近，并且每个模块相较于上一个模块相似度的上升幅度不是相同的，最后一个模块相对于上一个模块相似度的提升最大。

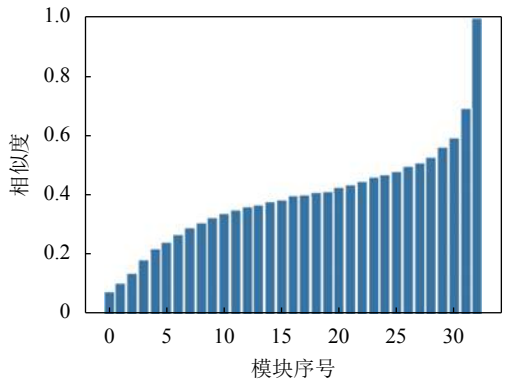


图 2　EDSR 模块相似性

前面提到超分网络学习到的是预测超分图像的残差，因此最后一个残差模块的输出可以认为是网络的最终学到的输出结果，那么每一个残差模块对于网络整体的贡献程度就可量化为相似度上升的幅度，即每个模块的相似度减去上一个模块的相似度就是该模块对于网络的贡献程度。图 3 为 EDSR 基于上述规则得到的模块贡献分布情况，横坐标代表每个残差模块的标号，纵坐标代表该模块对网络的贡献程度，图中已将模块贡献度映射到 0.0～1.0。

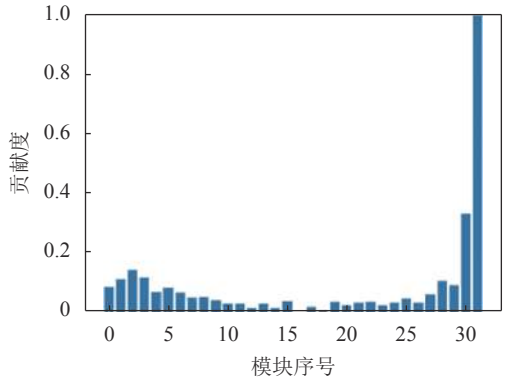


图 3　EDSR 模块贡献分布

得到每个模块对网络的贡献以后，按照贡献的高低对模块的重要性进行划分，低贡献度的模块重要性低，可以在模块剪枝中优先移除；而高贡献的模块重要性高，为了保证剪枝后的网络精度应予以保留。

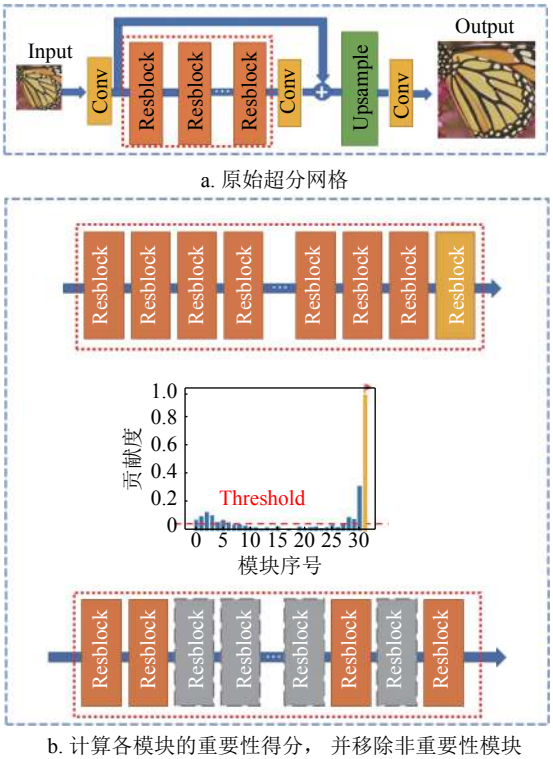
2.2　超分模块剪枝的步骤

2.2.1　数学标号

假设一个预训练的超分网络拥有 n 个残差模块，第 i 个残差模块表示为RB_i，它所占参数量为 W_{RB_i}。因为每个残差模块是完全相同的，因此每个模块的参数量也相同，即$W = W_{\mathrm{RB}_i}$。模型的总参数量则为$W_{\mathrm{total}} = nW + W_{\mathrm{other}}$，其中$W_{\mathrm{other}}$为除去残差模块外其他模块的参数量。通常$W_{\mathrm{other}}$在超分网络总参数量中仅占到很小一部分，因此$W_{\mathrm{total}} \approx nW$。另外$\mathrm{RB}_i$的输出用$\boldsymbol{O}_i$来表示，实验中输入 N 张图像，$\boldsymbol{O}_i^j$则代表了RB_i在输入第 j 张图像时输出的图像。当输入图像的高宽分别为h_j、w_j时，$\boldsymbol{O}_i^j$可用一个一维的向量表示，即$\boldsymbol{O}_i^j = \left[o_1^i, o_2^i, \cdots, o_k^i, \cdots, o_{c_i h_j w_j}^i\right]$，其中$c_i$表示输出$\boldsymbol{O}_i$的通道数，$o_k^i$代表$\boldsymbol{O}_i$一维向量的第$k$个元素。

2.2.2　超分模块剪枝步骤

基于相似度的超分网络剪枝步骤如图 4 所示。其中图 4a 为原始的超分网络，以 EDSR 为例，其超分网络由浅层特征提取模块、多个残差模块以及一个上采样模块组成。超分模块剪枝作用的主体是红色虚线框出来的残差模块部分。



a. 原始超分网格

b. 计算各模块的重要性得分， 并移除非重要性模块

图 4　超分模块剪枝的步骤

在图 4b 中，单独拿出总数为n的残差模块，首先输入一张图像，得到每个模块的输出$\boldsymbol{O}=\{\boldsymbol{O}_1,\boldsymbol{O}_2,\cdots,\boldsymbol{O}_i,\cdots,\boldsymbol{O}_n\}$，并以最后一个残差模块$\mathrm{RB}_n$的输出$\boldsymbol{O}_n$作为标准，计算其他每个模块的输出$\boldsymbol{O}_i$与$\boldsymbol{O}_n$的余弦相似度 (cosine similarity)：

$$\cos(\boldsymbol{O}_i,\boldsymbol{O}_n)=\frac{\sum_{k=1}^{c_ih_jw_j}\left(o_k^io_k^n\right)}{\sqrt{\sum_{k=1}^{c_ih_jw_j}\left(o_k^i\right)^2}\sqrt{\sum_{k=1}^{c_ih_jw_j}\left(o_k^n\right)^2}} \tag{1}$$

那么对于每个$\boldsymbol{O}_i$，相对于$\boldsymbol{O}_n$的相似度S_i便可表示为：

$$S_i=\cos(\boldsymbol{O}_n,\boldsymbol{O}_i) \tag{2}$$

为了消除单张图像输入带来的误差，本文在实际剪枝中使用了 N 张图像输入计算相似度，再取平均的方式来得到平均相似度，因此式 (2) 变成：

$$S_i=\frac{\sum_{j=0}^{N}\cos\left(\boldsymbol{O}_n^j,\boldsymbol{O}_i^j\right)}{N} \tag{3}$$

如 2.1 节所述，每个模块的贡献度可以量化为相似度的上升幅度，相似度上升的幅度越大，代表该模块对网络整体的贡献度越大，用贡献度等同于模块的重要性，可将模块重要性IMP_i表示为：

$$\mathrm{IMP}_i=S_i-S_{i-1} \tag{4}$$

在得到每个模块的重要性后，再根据设定的剪枝率，筛选不重要的模块进行模块剪枝，得到如图 4b 中所示的结构，其中灰色的部分即为被剪掉的不重要的模块。在剪掉不重要的模块后，网络的参数量以及 FLOPs 都得到了降低，降低的参数量可用mW来表示，m为被剪掉的残差模块数量。剪枝后的模型可以更加轻松地向下部署到低性能平台。但由于网络的结构已经发生改变，需要对模型进行微调恢复损失的精度。

综上，基于模块相似性的超分网络剪枝的算法如下。

算法 1 基于模块相似性的超分网络剪枝算法

输入：预训练完成的超分网络，包含n个残差模块$\mathrm{RB}=\{\mathrm{rb}_1,\mathrm{rb}_2,\cdots,\mathrm{rb}_n\}$，参数量为$nW+W_{\mathrm{other}}$；$N$张用于预测的低分辨率图像$\mathrm{LR}=\left\{\mathrm{lr}_1,\mathrm{lr}_2\cdots,\mathrm{lr}_j,\cdots,\mathrm{lr}_N\right\}$，每张图像的高宽分别为$h_j$、$w_j$；剪枝率thr。

输出：剪枝后的超分网络，包含$n-m$个残差模块，参数量为$(n-m)W+W_{\mathrm{other}}$。

1) 初始化模型权重

2) 评估模块重要性

for $j=1,2,\cdots,N$ do

以lr_j作为输入，得到每个残差模块的输出$\boldsymbol{O}_{\mathrm{RB}}=\{\boldsymbol{O}_1,\boldsymbol{O}_2,\cdots,\boldsymbol{O}_n\}$

for $i=1,2,\cdots,n$ do

计算$\boldsymbol{O}_i$与$\boldsymbol{O}_n$的余弦相似度，$S_i^j=\cos(\boldsymbol{O}_i,\boldsymbol{O}_n)$

end for

计算平均相似度$S_i^j=\dfrac{S_i^{j-1}+S_i^j}{2}$

end for

计算每个模块的重要性，$\mathrm{IMP}_i=S_i-S_{i-1}$

3) 模块剪枝

if $\mathrm{IMP}_i<\mathrm{thr}$ then

移除rb_i

end if

4) 模型微调

关于剪枝率thr的取值，应依据希望移除的模块数量来决定。thr越大，移除的模块数量越多，同时可能造成的精度损失也越大。然而thr的大小与剪枝后精度损失的大小并没有绝对关系，应依据待剪枝模型的冗余程度而定。

3 实验结果及分析

3.1 实验配置

为了验证剪枝方法，本文选择了代表性的超分网络 EDSR 进行实验，同时为了证明方法的通用性，也在 RCAN 进行了剪枝实验。预训练和 Fine-tune 阶段以 DIV2K[39] 作为数据集。DIV2K 是一个有 900 张 2 k 分辨率图像的数据集，实验中取 1～800 张作为训练集，801～900 张的图像作为验证集。测试部分则是选择了 Set5[40]，Set14[41]，BSD100[42]，Urban100[43] 这 4 个 benchmarks 数据集。通过计算 YCbCr 中 Y 通道的峰值信噪比 (PSNR) 来量化超分实验结果。Fine-tune 阶段使用了 Adam 作为优化器，学习率设置为 1×10^{-4}，每 200 个 epoch 学习率衰减为原来的一半，使用L_1作为损失函数，训练 300 个 epoch。硬件方面则使用了 NVIDIA RTX 2070 作为训练 GPU。

3.2 实验结果与分析

3.2.1 超分模块剪枝效果

EDSR 原模型拥有 32 个残差模块，参数量共

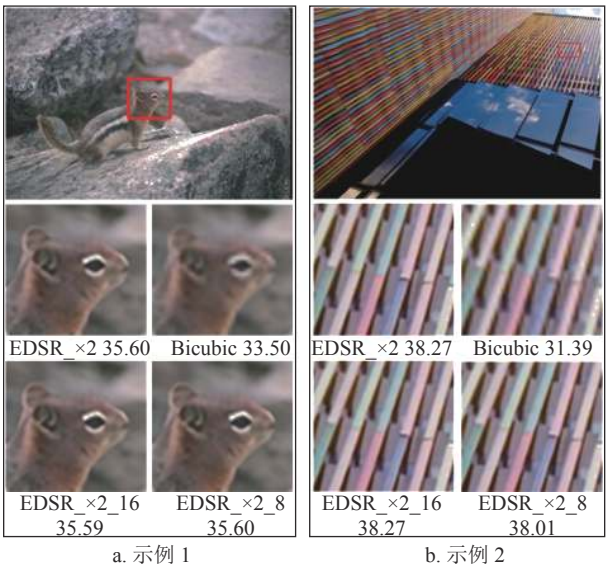
为 37.76 M，占到整个模型的 92.7%，因此直接减少残差模块的数量是一种行之有效的压缩模型的方法。实验中分别将残差模块剪到 16 个和 8 个。而 RCAN 则拥有 10 个残差模块组，每组又拥有 20 个残差模块，总模块数达到了 200 个，因此也非常适合模块剪枝。在对 RCAN 剪枝的过程中，对每个残差模块组的模块都进行了剪枝，每组剪去等量数目的模块。最终得到的剪枝结果如表 1 所示。

表 1　超分模块剪枝的定量结果

Model	Params×10^6	Flops×10^9	PSNR/ dB，SSIM			
			Set5	Set14	BSD100	Urban100
EDSR_×2	40.73	2671	38.19，0.9601	33.94，0.9193	32.36，0.9011	32.97，0.9351
EDSR_×2_16	21.85	1433	38.16，0.9601	33.81，0.9188	32.29，0.9004	32.62，0.9322
EDSR_×2_8	**12.41**	**814**	38.06，0.9598	33.73，0.9181	32.23，0.8994	32.33，0.9292
RCAN_×2	15.44	1005	38.27，0.9606	34.11，0.9208	32.41，0.9018	33.34，0.9374
RCAN_×2_6	**5.02**	**327**	38.19，0.9603	33.85，0.9191	32.31，0.9006	32.85，0.9336

EDSR_×2_16/8 分别代表剪枝后拥有 16 个残差模块和 8 个残差模块的 EDSR，而 RCAN_×2_6 则代表每个残差模块组拥有 6 个残差模块，因此 RCAN 总模块数由 10×20 个减少到了 10×6 个。Flops 使用1×3×256×256的输入测得。PSNR 代表超分图像与原图之间的峰值信噪比，是一种评价图像失真水平的客观标准，PSNR 的值越高代表图像质量越高。SSIM 代表结构相似性，也是一种评判图像相似度的指标，SSIM 越高代表图像越相似，完全一致时 SSIM 为 1.0。从表格中可以看到 EDSR 模型参数量分别下降了 46% 和 69%，然而在测试数据集上的平均 PSNR 却只有轻微下降，在 RCAN 的实验中也是如此。由此可见，超分重建图像的质量不会造成太大损失。

为了更直观地展现超分模块剪枝对模型精度的影响，本文提供了可视化的测试结果，如图 5 所示。



a. 示例 1　　b. 示例 2

图 5　可视化测试结果

图中 EDSR_×2 代表原模型，EDSR_×2_16/8 代表剪枝后的模型，Bicubic 代表双三次插值算法得到的插值图像。本文分别从 BSD100 和 Urban100 两个测试集中挑选一张图片作为可视化测试结果，可以看到剪枝后的模型不仅从 PSNR 上非常接近原模型，从可视化的结果来看也十分接近原模型的超分结果，即便是将模型剪枝到了 8 个模块，超分的效果也远好于插值图像。

3.2.2　超分模型剪枝的性能提升

超分模块剪枝对模型的压缩能力不仅体现在参数量和运算量的减少，同时提升了模型的预测速度。为了测试剪枝后模型的预测速度，本文使用一个1×3×256×256的输入，测试在 GPU 上预测 10 次所需要的时间，测试时使用的 GPU 为 NVIDIA RTX 2070 显卡。测试结果如表 2 所示。

表 2　超分模块剪枝对 EDSR 预测速度的提升

Model	预测时间/s	GPU显存占用/MB
EDSR_×2	5.11	834.79
EDSR_×2_16	2.73	759.26
EDSR_×2_8	**1.58**	**721.50**

表中 GPU 显存占用为 Pytorch 提供的显存占用查看方法 torch.cuda.max_memory_allocated() 测得。可见超分模块剪枝可以有效提升模型的预测速度，其中 8 模块的剪枝模型预测速度达到了 3 倍以上的提升。不仅如此，模型的显存占用同样得到了一定程度的节约，这对于实际部署到低性能平台很有意义。

3.3　消融实验

为了进一步说明基于模块相似的超分模块剪枝的有效性和合理性，本文首先验证超分模块剪枝的有效性，其次证明超分模块之间的相对独立性，并比较了不同的相似性度量方法，同时和知识蒸馏方法进行了对比。

3.3.1 超分模型剪枝的有效性验证

为了验证基于模块相似性的超分模块剪枝的有效性，本文设计了以下实验：1) 使用超分模块剪枝的方法计算模块重要性，并依据模块重要性进行网络剪枝；2) 不计算模块重要性，随机选择残差模块进行剪枝；3) 使用超分模块剪枝，在剪枝后对模型参数重新初始化，不使用预训练模型的参数。以上 3 种实验均使用同样的剪枝率，保留 8 个残差模块，Fine-tune 的所有步骤保持一致。分别用 Pruned、Random、Scratch 表示这 3 种方法，得到表 3 的实验结果。

表 3 超分模块剪枝有效性实验结果

Model	Block Num	PSNR/dB
EDSR_x2	32	35.03
Scratch	8	34.74
Random	8	34.75
Pruned	8	**34.78**

从验证集上的 PSNR 表现来看，Pruned 方法优于 Random 方法，这说明模块重要性的评估方法是有效的，通过本文方法可以有效分辨出对网络整体贡献更高、更重要的残差模块，从而做到精准的网络剪枝。同时，Pruned 方法和 Random 方法都优于 Scratch 方法，这说明使用了网络剪枝得到的网络相比于从头开始训练的网络能取得更高的精度表现，证明了网络剪枝本身的有效性。

3.3.2 模块间的相对独立性验证

为了验证模块的相对独立性，本文设计了以下实验。本实验同样采用了 EDSR 网络作为样本，首先使用模块重要性评估方法选出模型中最重要以及最不重要的两个模块，使用两个相同的 EDSR 模型，移除最重要的模块，再则移除最不重要的模块。两个模型在相同的条件下进行 Fine-tune 恢复精度，并记录下前 10 个 epoch 在验证集上的 PNSR 变化情况，得到的结果如图 6 所示。

在超分网络中的不同模块是相互联系而不是完全独立的。尽管如此，每个网络模块也不是完全平等的，而是具有相对的独立性。这表现在去掉不同的模块对模型输出的影响并不相同，有的模块会剧烈地影响模型的输出，有的只是轻微影响。对于本文剪枝方法而言，就是要找出对于整个网络而言相对不重要，影响轻微的模块，并移除该模块，以达到快速恢复模型精度的目的。

正如实验中所示，移除重要性相对低的模块在 Fine-tune 过程中模型精度上升的速度远大于另一个，证实了不重要模块对模型整体的影响更小，也证明了本文重要性评估的合理性。

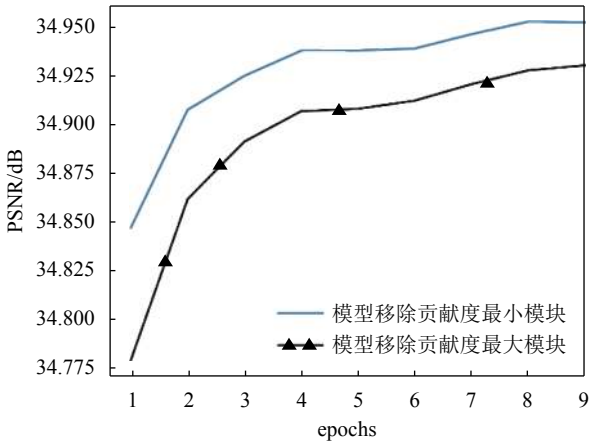


图 6 模块的相对独立性验证

3.3.3 不同相似性计算方法对比

在评估模块重要性的算法中，本文使用了余弦相似度作为计算相似度的方法。使用均方误差 (MSE) 作为相似度的指标进行了表 4 中的实验。

表 4 不同的相似度计算方法对比

方法	PSNR/dB
均方误差	34.72
余弦相似度	34.78

和前面的实验相同，除了相似度计算的方法，其他条件均保持一致的情况下。在剪到 8 个残差模块后，使用余弦相似度的方法相较于均方误差能够取得更好的精度表现，因此最终选择了余弦相似度作为相似度的指标。

3.3.4 与知识蒸馏的方法对比

对比文献 [37] 中知识蒸馏的方法，本文的模块剪枝更具有导向性，在 RCAN 上进行相同数量模块的剪枝，对比结果如表 5 所示。可知在同等模块数量的情况下，本文方法在不同的数据集上都取得了更佳的精度。

表 5 RCAN 上不同数据集 PSNR 对比

dB

方法	Block Num	Set5	Set14	BSD100	Urban100
文献[37]	6	38.16	33.81	32.27	32.53
本文	6	**38.19**	**33.85**	**32.31**	**32.85**

4 结 束 语

本文提出了一种适用于常见超分网络的模块剪枝方法，通过计算模块的相似性换算得到网络中每

个模块对整体网络的贡献程度，并且通过移除贡献度低的模块达到网络剪枝的目的。

相比于粒度更低的权重剪枝以及通道剪枝，本文的模块剪枝属于层剪枝范畴，操作更加灵活方便，并且在不同的网络上都取得了良好的剪枝效果，为超分网络在低性能平台上的部署提供了可能。

参 考 文 献

[1] CABALLERO J. Cardiac image super-resolution with global correspondence using multi-atlas patchmatch[J]. Med Image Comput Comput Assist Interv, 2013, 16(3): 9-16.

[2] DAI D, WANG Y, CHEN Y, et al. Is image super-resolution helpful for other vision tasks?[C]//2016 IEEE Winter Conference on Applications of Computer Vision (WACV). Piscataway: IEEE, 2016: 1-9.

[3] SAJJADI M, SCHOLKOPF B, HIRSCH M. EnhanceNet: Single image super-resolution through automated texture synthesis[C]//IEEE International Conference on Computer Vision. Piscataway: IEEE, 2017: 4491-4500.

[4] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[J]. Advances in Neural Information Processing Systems, 2012, 25(2): 1097-1105.

[5] DONG C, LOY C C, HE K, et al. Image super-resolution using deep convolutional networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 38(2): 295-307.

[6] YANG J, WRIGHT J, HUANG T, et al. Image super-resolution as sparse representation of raw image patches[C]//2008 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2008: 1-8.

[7] YANG J, WRIGHT J, HUANG T S, et al. Image super-resolution via sparse representation[J]. IEEE Transactions on Image Processing, 2010, 19(11): 2861-2873.

[8] LEDIG C, THEIS L, HUSZÁR F, et al. Photo-realistic single image super-resolution using a generative adversarial network[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2017: 4681-4690.

[9] LIM B, SON S, KIM H, et al. Enhanced deep residual networks for single image super-resolution[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. Piscataway: IEEE, 2017: 136-144.

[10] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2016: 770-778.

[11] ZHANG Y, LI K, LI K, et al. Image super-resolution using very deep residual channel attention networks[C]//Proceedings of the European Conference on Computer Vision (ECCV). Munich: ECCV, 2018: 286-301.

[12] COURBARIAUX M, HUBARA I, SOUDRY D, et al. Binarized neural networks: Training deep neural networks with weights and activations constrained to+1 or-1[EB/OL]. [2021-03-15]. https://arxiv.org/abs/1602.02830.

[13] JACOB B, KLIGYS S, CHEN B, et al. Quantization and training of neural networks for efficient integer-arithmetic-only inference[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2018: 2704-2713.

[14] LI H, YAN C, LIN S, et al. PAMS: Quantized super-resolution via parameterized max scale[EB/OL]. [2021-03-15]. https://arxiv.org/abs/2011.04212.

[15] HINTON G, VINYALS O, DEAN J. Distilling the knowledge in a neural network[EB/OL]. [2021-03-20]. https://arxiv.org/abs/1503.02531.

[16] YIM J, JOO D, BAE J, et al. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2017: 4133-4141.

[17] GAO Q, ZHAO Y, LI G, et al. Image super-resolution using knowledge distillation[C]//Asian Conference on Computer Vision. Cham: Springer, 2018: 527-541.

[18] LI H, KADAV A, DURDANOVIC I, et al. Pruning filters for efficient convnets[EB/OL]. [2021-03-18]. https://arxiv.org/abs/1608.08710.

[19] GAO S, HUANG F, CAI W, et al. Network pruning via performance maximization[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2021: 9270-9280.

[20] HE Y, ZHANG X, SUN J. Channel pruning for accelerating very deep neural networks[C]//Proceedings of the IEEE International Conference on Computer Vision. Piscataway: IEEE, 2017: 1389-1397.

[21] LIN S, JI R, YAN C, et al. Towards optimal structured cnn pruning via generative adversarial learning[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2019: 2790-2799.

[22] HOU Z, KUNG S Y. Efficient image super resolution via channel discriminative deep neural network pruning[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Piscataway: IEEE, 2020: 3647-3651.

[23] KIM J, LEE J K, LEE K M. Accurate image super-resolution using very deep convolutional networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2016: 1646-1654.

[24] ZHANG Y, TIAN Y, KONG Y, et al. Residual dense network for image super-resolution[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2018: 2472-2481.

[25] CHEN Y, GUO B, SHEN Y, et al. Using efficient group pseudo-3D network to learn spatio-temporal features[J]. Signal, Image and Video Processing, 2021, 15(2): 361-369.

[26] HOWARD A G, ZHU M, CHEN B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[EB/OL]. [2021-03-15]. https://arxiv.org/abs/

1704.04861.
[27] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions[EB/OL]. [2021-03-15]. https://arxiv.org/abs/1409.4842.
[28] DENIL M, SHAKIBI B, DINH L, et al. Predicting parameters in deep learning[EB/OL]. [2021-04-05]. https://arxiv.org/abs/1306.0543.
[29] HAN S, POOL J, TRAN J, et al. Learning both weights and connections for efficient neural networks[EB/OL]. [2021-03-12]. https://arxiv.org/abs/1506.02626.
[30] HAN S, MAO H, DALLY W J. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding[EB/OL]. [2021-03-15]. https://arxiv.org/abs/1510.00149.
[31] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84-90.
[32] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[EB/OL]. [2021-03-15]. https://arxiv.org/abs/1409.1556.
[33] LI H H, KADAV A, DURDANOVIC I, et al. Pruning filters for efficient convnets[EB/OL]. [2021-03-15]. https://arxiv.org/abs/1608.08710.
[34] 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. Piscataway: IEEE, 2017: 2736-2744.
[35] CHEN S, ZHAO Q. Shallowing deep networks: Layer-wise pruning based on feature representations[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 41(12): 3048-3056.
[36] ALAIN G, BENGIO Y. Understanding intermediate layers using linear classifier probes[EB/OL]. [2021-03-12]. https://arxiv.org/abs/1610.01644.
[37] HE Z, DAI T, LU J, et al. Fakd: Feature-affinity based knowledge distillation for efficient image super-resolution[C]//2020 IEEE International Conference on Image Processing (ICIP). Piscataway: IEEE, 2020: 518-522.
[38] DAI T, CAI J, ZHANG Y, et al. Second-order attention network for single image super-resolution[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2019: 11065-11074.
[39] AGUSTSSON E, TIMOFTE R. Ntire 2017 challenge on single image super-resolution: Dataset and study[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. Piscataway: IEEE, 2017: 126-135.
[40] BEVILACQUA M, ROUMY A, GUILLEMOT C, et al. Low-complexity single-image super-resolution based on nonnegative neighbor embedding[J]. British Machine Vision Conference, 2012, 135(1): 1-10.
[41] ZEYDE R, ELAD M, PROTTER M. On single image scale-up using sparse-representations[C]//International Conference on Curves and Surfaces. Berlin, Heidelberg: Springer, 2010: 711-730.
[42] MARTIN D, FOWLKES C, TAL D, et al. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics[C]//Proceedings Eighth IEEE International Conference on Computer Vision. [S.l.]: IEEE, 2001: 416-423.
[43] HUANG J B, SINGH A, AHUJA N. Single image super-resolution from transformed self-exemplars[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2015: 5197-5206.

编 辑 税 红