基于高效的多尺度特征提取的轻量级语义分割

刘 云[1] 陆承泽[1] 李仕杰[2] 张 乐[3] 吴宇寰[1] 程明明[1]

[1](南开大学计算机学院 天津 300350)

[2](波恩大学信息系统与人工智能系 波恩 53115 德国)

[3](新加坡科技研究局 新加坡 138632 新加坡)

摘 要 近来移动端视觉应用的发展激发了对轻量级语义分割技术的需求.尽管取得了十分辉煌的成就,当前轻量级语义分割模型仍存在精度不足、参数过多的问题.本文的目的在于开发一个具有少量参数的高精度分割模型.为此,本文基于以下观察提出了一种新的轻量级分割模型 MiniNet:(1)语义分割依赖于多尺度特征学习;(2)下采样是加速网络推理和扩大卷积感受野的最有效方法;(3)网络深度和卷积通道数之间的良好平衡对于轻量级模型至关重要.具体来说,MiniNet 采用空间金字塔卷积(Spatial Pyramid Convolution,SPC)模块和空间金字塔池化(Spatial Pyramid Pooling,SPP)模块作为多尺度特征学习的基本单元.此外,MiniNet 将大多数网络层和操作放在较小的尺度上,即原始图像分辨率的 1/16,而不是先前模型中常用的 1/8 尺度.MiniNet 还设法平衡网络深度和卷积通道数.在没有 ImageNet 预训练的情况下,MiniNet 在 Cityscapes 测试数据集上仅以 211K 参数和 94.3fps 的速度即可达到 66.3%的 mIoU.

关键词 语义分割;轻量级语义分割;快速语义分割;图像分割;轻量级网络

中图法分类号 TP391 **DOI 号** 10.11897/SP.J.1016.2022.01517

Efficient Multi-Scale Feature Extraction for Lightweight Semantic Segmentation

LIU Yun[1] LU Cheng-Ze[1] LI Shi-Jie[2] ZHANG Le[3] WU Yu-Huan[1] CHENG Ming-Ming[1]

[1](*College of Computer Science, Nankai University, Tianjin* 300350)

[2](*Department of Information Systems and Artificial Intelligence, Bonn University, Bonn* 53115, *Germany*)

[3](*Agency for Science Technology and Research, Singapore* 138632, *Singapore*)

Abstract Recent interest in many mobile vision applications has generated a high demand for lightweight semantic segmentation. Despite glorious achievements, current lightweight models suffer from either unsatisfactory accuracy or too many parameters. The goal of this paper is to develop an accurate segmentation model with a small number of parameters. To this end, we propose a new lightweight segmentation model, namely MiniNet, based on several observations: (1) semantic segmentation depends on multi-scale learning; (2) downsampling is the most effective way to speed up network inference and enlarge the receptive fields; (3) a good balance between network depth and the number of convolution channels is essential for lightweight models. Specifically, MiniNet adopts spatial pyramid convolution (SPC) and spatial pyramid pooling (SPP) modules as the basic units for multi-scale feature learning. Besides, MiniNet puts most layers and operations on a small scale, i.e., 1/16 of the original image resolution, rather than 1/8 scale in the previous models. MiniNet also manages to balance the network depth and the

收稿日期:2020-10-31;在线发布日期:2021-09-07.本课题得到新一代人工智能重大项目(2018AAA0100400)、国家自然科学基金优秀青年科学基金项目(61922046)、教育部指导高校科技创新规划项目资助.**刘 云**,博士研究生,中国计算机学会(CCF)会员,主要研究方向为计算机视觉.E-mail:vagrantlyun@gmail.com.**陆承泽**,硕士研究生,主要研究方向为计算机视觉.**李仕杰**,博士研究生,主要研究方向为计算机视觉.**张 乐**,博士,研究科学家,主要研究方向为深度学习.**吴宇寰**,博士研究生,主要研究方向为计算机视觉.**程明明**(通信作者),博士,教授,中国计算机学会(CCF)高级会员,主要研究领域为计算机视觉、计算机图形学.E-mail:cmm@nankai.edu.cn.

number of convolution channels. Without ImageNet pretraining, MiniNet achieves 66.3% mIoU on the Cityscapes test dataset with only 211K parameters and speed of 94.3 fps.

Keywords semantic segmentation; lightweight semantic segmentation; efficient semantic segmentation; image segmentation; lightweight network

1 引 言

语义分割是计算机视觉中的一个基本问题. GPU 不断增长的计算能力加速了用于精确语义分割的全卷积网络(Fully Convolutional Network, FCN)的发展. 对于最新的模型[1-8],通过引入更多的参数和各种复杂的操作来提高精度是很常见的. 例如,PSPNet[1]的参数约为 66M,它需要几秒钟的时间才能在 TITAN Xp GPU 上处理一张普通图像. 但是,例如机器人、智能手机、自动驾驶汽车和增强现实智能眼镜这样的移动设备无法部署大型且耗电的强大 GPU(例如,TITAN Xp GPU 的功耗约为 250 W),因此有限的计算资源阻止了最新的分割模型[9-10]的实际应用. 此外,移动设备只有有限的存储空间. 例如,智能手机不可能使用数百 MB 的内存来存储针对某个特定应用的预训练的深度模型. 这启发我们开发在准确性、效率、参数量和功耗之间取得良好平衡的语义分割模型.

为此,研究者们近来对轻量级语义分割的研究兴趣迅速增加,已经出现了许多轻量级的分割模型[9-17]. 这些模型通常采用深度可分离卷积[10,14-15,17]、非对称卷积[11-12,16]和密集连接[16]等技术,以减少网络参数和操作量. 为了用浅层网络获得较大的感受野,在轻量级模型中还使用了扩张卷积[18]. 尽管现有技术水平已得到一定程度的发展,但当前模型要么无法取得令人满意的准确率,要么参数量过多. 例如,ESPNetv2[10]具有 0.73M 参数,但在 Cityscapes 测试数据集[19]上仅达到 62.1%的 mIoU;而 ICNet[13]达到了 69.5%的 mIoU,却具有 6.68M 参数(即需要 6.68M×4=26.72M 的存储内存). 为了使模型能灵活地应用于移动设备,本文认为参数数量应少于 0.5M,即少于 0.5M×4=2.0M 的存储空间.

在介绍所提出的模型之前,本文总结了一些对语义图像分割任务的观察. 首先,语义分割高度依赖于多尺度学习来识别自然场景中的多尺度物体,这是以往的优秀方法成功的关键[20-21]. 现有的研究已经提出了各种技术来利用多尺度的深度学习特征,例如编码-解码[21-22]结构、空洞空间金字塔池化(Atrous Spatial Pyramid Pooling, ASPP)[23]、金字塔池化模块(Pyramid Pooling Module)[1]和多路径优化(Multi-path Refinement)[2]. 其次,降低特征图的分辨率是提高推理速度和扩大感受野的最有效方法. 例如,一半分辨率下的特征图所需的操作数是原始特征图的操作数的 1/4. 因此,最近非常深的神经网络[24-25]通常会将输入图像降采样到非常小的分辨率. 最后,在给定一定数量的参数的情况下,与具有更多卷积通道数的较浅的网络相比,具有较少卷积通道数的适当的较深的网络可以实现更高的精度,但速度更低. 例如,假设一个具有 C 个输入通道和 C 个输出通道的 1×1 卷积,其参数的数量为 C^2. 如果将输入和输出通道的数量更改为 $C/2$,则参数的数量变为 $C^2/4$. 因此,好的轻量级分割模型应在网络深度和卷积通道数之间做出良好的权衡,以实现更好的性能. 由于模型较小,MiniNet 不需要 ImageNet[26]预训练,因此可以灵活地适应新数据和新任务.

本文在三个数据集上进行了广泛的实验,包括 Cityscapes[19]、CamVid[27]和 Mapillary Vistas[28],以证明所提出的 MiniNet 的有效性和高效率. 在 Cityscapes 测试数据集[19]上,没有 ImageNet[26]预训练的情况下,MiniNet 仅用 211K 参数和 2.4G FLOPs 达到了 66.3%的 mIoU,速度达到了 94.3fps. 较小版本的 MiniNet 以 95K 参数量,能够以 104.2fps 的速度达到 64.1%的 mIoU. 本文还进行了详细的消融实验,以评估各种设计选择的影响.

2 相关工作

由于学术和工业上的广泛用途,语义分割是一个热门话题[6-8,29-33]. 自全卷积网络(FCN)[21]发明以来,基于 FCN 的深度学习方法就统治了语义分割领域. 本节先简要回顾经典的高精度的语义分割模型和技术,然后概述轻量级的模型.

多尺度学习. 自然场景中的物体呈现出很大的尺度变化,因此多尺度学习对于语义分割至关重要. 大多数方法旨在设计网络以从彩色图像中学习

有效的多尺度特征表示. 例如，FCN[21]、U-Net[22]、DeconvNet[34]和 SegNet[35]建立了编码-解码网络，从而以一种从顶层到低层的方式来融合深度特征. 一些方法[2,20,36-37]聚合来自多层的多尺度深度特征，以进行最终的密集预测. DeepLab[23]及其变体[4,38-39]通过使用具有不同扩张率的扩张卷积来设计 ASPP 模块，以学习多尺度特征. 基于 ASPP 模块，DenseASPP[40]以密集的方式连接一组扩张卷积，从而生成密集地覆盖了更大尺度范围的多尺度特征.

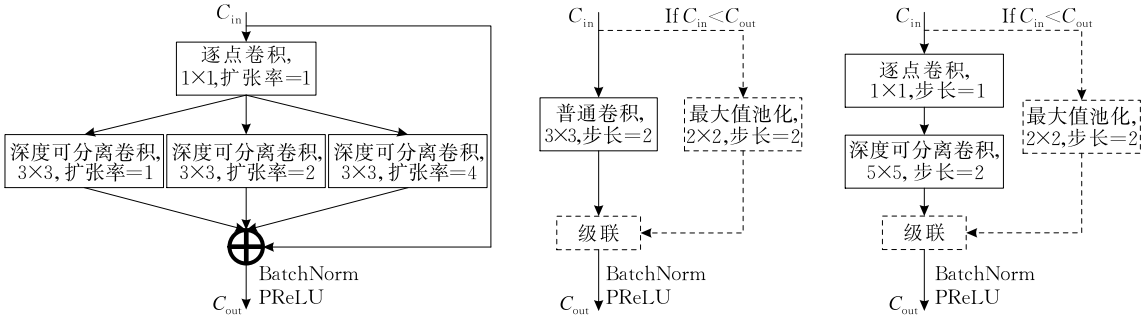
全局上下文信息. 除了多尺度学习之外，一些研究还致力于通过上下文编码[3]、金字塔池化[1]和非近邻操作[41-42]来利用全局上下文信息. 此外，Wu 等人[43]试图找到网络深度和宽度之间的良好折衷，以提高分割精度. DFN[5]通过设计网络来处理类内不一致问题，并引入了一个边界网络来使边界两侧的特征可区分. 一些方法[38,44-45]使用条件随机场(Conditional Random Field，CRF)或马尔可夫随机场(Markov Random Field，MRF)来建模语义分割中的空间关系. 上述模型旨在不考虑模型大小和推理速度的情况下提高分割精度，因此对于移动设备来说并不适用. 本文的目标是设计一种网络规模小、速度快、准确性高的轻量级模型.

轻量级语义分割. ENet[46]打开了轻量级语义分割的大门. 它减少了 ResNet[24]的卷积通道，以少量参数实现实时分割. ERFNet[11]将标准的二维卷积分解为两个非对称的一维卷积. ContextNet[15]结合了小分辨率的深层网络和全分辨率的浅层网络. ESPNet[9]将标准卷积分解为一个逐点卷积和由扩张卷积构成的空间金字塔. ESPNetv2[10]在 ESPNet[9]的基础上进行了扩展，它使用分组的逐点卷积和深度可分离的扩张卷积. ICNet[13]、BiSeNet[14]、SQNet[47]和 FRRN[48]试图在分割精度和推理速度之间取得良好的平衡. 最近的一些技术报告[16-17]也为轻量级语义分割提供了新的设计. 本文的目标是在不牺牲速度和增加参数量的情况下提高轻量级分割的准确性.

3　方　法

空间金字塔卷积. 众所周知，一个标准卷积可以分解为一个逐点卷积和一个深度可分离卷积[49]. 逐点卷积实际上就是 1×1 卷积，深度可分离卷积是分组卷积，其分组数等于输出通道数. 如上所述，多尺度学习对于语义分割至关重要. 为了有效地进行多尺度学习，SPC 模块用一组扩张金字塔卷积代替了单个深度可分离卷积. 假设有 r 个并行的扩张卷积，则其扩张率分别为 $1,2,\cdots,2r-1$. 那么 SPC 模块能够很自然地学习到多尺度信息，其感受野分别为 $3,5,\cdots,2r+1$，其中最大的感受野为 $2r+1$，远大于标准卷积. 为了更好的训练优化，本文在将多个并行的扩张卷积的结果逐元素相加之后，添加了一个残差连接[24]，其后使用批归一化[50]和非线性激活 PReLU[51]. 图 1(a)中显示了一个具有三个分支的 SPC 模块.

空间金字塔池化. SPP 模块用于对编码路径中的特征图进行下采样. 它由两个分支组成，其中一个分支在不重叠的 2×2 窗口内进行最大值池化，另一个分支使用标准卷积或分解卷积，这两个版本分别如图 1(b)和图 1(c)所示. 在第一个卷积阶段中，SPP 使用标准卷积，其通道数较小，随后具有分解卷积的 SPP 在更深层被使用. 假设有 C_{in} 个输入通道和 C_{out} 个输出通道. 若 $C_{in}<C_{out}$，则最大值池化分支会生成一个 C_{in} 个通道的特征图，卷积分支负责生成输出特征图中剩下的 $C_{out}-C_{in}$ 个通道. 否则，最大值池化分支会被省略，卷积分支会直接生成一个通道数为 C_{out} 的特征图. 本文将在第 4.2 节中展示这种二分支设计优于单个分支的设计.



(a) 带有三个分支的SPC模块　(b) 带有标准卷积的SPP模块　(c) 带有深度可分离卷积的SPP模块

图 1　MiniNet 中基本模块的图示(图中，C_{in} 表示输入特征图的通道数，而 C_{out} 表示输出特征图的通道数)

尺度下采样. 记 $F_{\text{in}}\in\mathbb{R}^{C_{\text{in}}\times Y\times X}$ 为一个卷积层的输入特征图，$F_{\text{out}}\in\mathbb{R}^{C_{\text{out}}\times Y'\times X'}$ 为输出特征图. 卷积核可以定义为 $T\in\mathbb{R}^{C_{\text{out}}\times C_{\text{in}}\times t\times t}$，其中 $t\times t$ 表示核大小. 输出特征图的大小由卷积步长 s(即 $Y'=Y/s$ 和 $X'=X/s$)控制. 因此，在这样的卷积层中，操作数大约与 $X'Y'C_{\text{out}}C_{\text{in}}t^2=XYC_{\text{out}}C_{\text{in}}t^2/s^2$ 成正比. 从这种表述中，可以发现卷积层中的运算数量大约与输入特征图、输入和输出通道数以及卷积核大小成正比. 如果将特征图下采样 2 倍，那么下采样后的特征图的操作数将仅具有原始操作数的 1/4. 此外，下采样还可以将感受野扩大两倍，从而可以减轻对网络深度的需求. 因此，减小图像大小是加速卷积网络的最有效方法. 最后，减小特征图尺度、增大感受野，也将有利于提高分割性能，因为很多研究表明增大感受野将有助于卷积神经网络进行语义识别[41-42,52]. 之前的语义图像分割模型通常仅将图像降采样为原图像的 1/8 大小[9,11-12,15-17]，但是在本文中，将大部分卷积操作放置在原图像的 1/16 尺度下.

网络结构. 图 2 中展示了所提出的 MiniNet 的网络结构. MiniNet 是一个具有编码-解码结构的卷积神经网络. 其中，编码部分由四个阶段组成. 因为低层网络层的通道数通常较少，所以标准卷积比深度可分离卷积更高效，参数也只是略多[49,53-54]. 因此，编码部分的第一个阶段使用标准卷积，而不是深度可分离卷积. 对于编码的第一阶段，首先使用带有标准卷积的 SPP 模块(如图 1(b)所示)将输入图像降采样到 1/2 的尺度下. 随后，将 P_1 个残差卷积模块顺序地连接，每个残差卷积模块可以用如下公式表示：

$$F_1^l=\text{PReLU}(\text{BatchNorm}(W_1^l * F_1^{l-1}+F_1^{l-1})) \quad (1)$$

其中，$*$ 表示卷积操作符，F_1^l 表示第一编码阶段的第 $l(l\in\{1,2,\cdots,P_1\})$ 个残差卷积模块的输出特征图. $W_1^l\in\mathbb{R}^{C_1\times C_1\times 3\times 3}$ 是第一阶段的第 l 个模块的权重，其中，C_1 是特征的通道数. SPP 模块的输出特征图是 F_1^0. 注意为了提高效率，本文遵循之前研究[12]的建议忽略了卷积的偏差项.

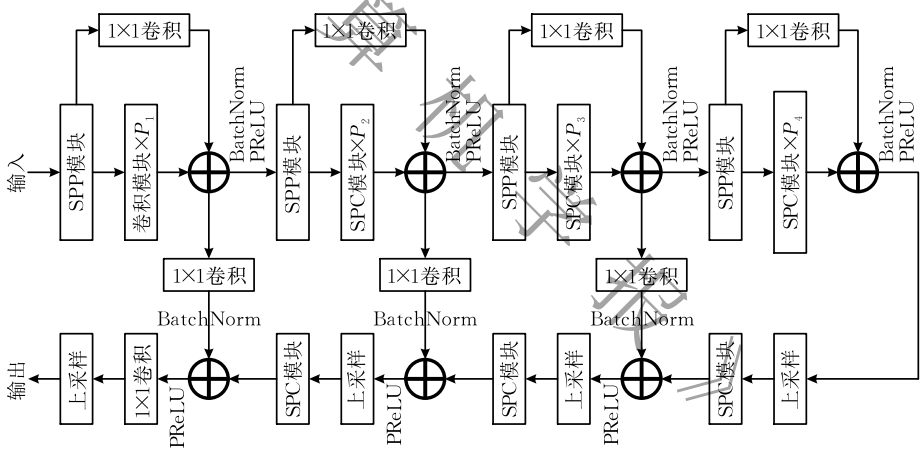


图 2 所提出的 MiniNet 网络结构

对于第二个编码阶段，本文首先应用具有深度可分离卷积的 SPP 模块(图 1(c))将特征图进一步下采样为 1/4 尺度. 然后，P_2 个 SPC 模块(图 1(a))紧随其后. 编码部分的第三个和第四个阶段与第二个阶段相似，分别将特征图下采样到 1/8 和 1/16 尺度. 与第一个阶段类似，对于后三个阶段，定义卷积通道数分别为 C_2、C_3、C_4，SPC 模块的数量分别为 P_2、P_3、P_4，输出特征图分别为 $F_2^l(l\in\{1,2,\cdots,P_2\})$、$F_3^l(l\in\{1,2,\cdots,P_3\})$、$F_4^l(l\in\{1,2,\cdots,P_4\})$. 后三个 SPP 模块的输出特征图分别为 F_2^0、F_3^0 和 F_4^0. 对于第 k 个阶段的 SPP 模块，池化分支能够生成带有 C_{k-1} 个通道数的特征图，卷积分支(标准卷积或深度可分离卷积)负责生成剩下的 (C_k-C_{k-1}) 个通道. 因此，第 k 个阶段的所有的残差卷积模块或 SPC 模块拥有 C_k 个输入特征通道和 C_k 个输出特征通道. 为了便于优化，本文对于编码部分每个阶段的 SPP 模块和最后一个 SPC 模块设计了一个长残差连接，可用如下公式表示：

$$F_k^{P_k}=\text{PReLU}(\text{BatchNorm}(\boldsymbol{W}_k^{\text{long}} * F_k^0+F_k^{P_k})),$$
$$\text{s.t. } k\in\{1,2,3,4\} \quad (2)$$

其中，$\boldsymbol{W}_k^{\text{long}}\in\mathbb{R}^{C_k\times C_k\times 1\times 1}$ 是第 k 个编码阶段的 1×1 卷积的权重矩阵. 除了长残差连接，本文将每个基础模块(即 SPC 模块或残差卷积模块)内部的残差连接称为短残差连接.

解码网络逐渐聚合顶部的粗糙的语义特征和底部的细粒度特征，以分割图像并使其具有清晰的边

界.解码网络共有三个阶段,每个阶段可以写成如下形式:

$$F_k^{D1}=\text{BatchNorm}(\boldsymbol{W}_k^D * F_k^{P_k}),$$
$$F_k^{D2}=\text{SPC}(\text{Upsampling}(F_{k+1}^D)),$$
$$F_k^D=\text{PReLU}(F_k^{D1}+F_k^{D2}),$$
$$\text{s.t. } k\in\{1,2,3\} \tag{3}$$

其中,$\boldsymbol{W}_k^D\in\mathbb{R}^{C_{k+1}\times C_k\times 1\times 1}$表示1×1卷积层的权重矩阵.注意,$F_4^D=F_4^{P_4}$.本文省略了式(3)中SPC模块的PReLU激活函数,因为PReLU随后已被用来激活F_k^{D1}和F_k^{D2}的和.最终,本文对F_1^D使用一个1×1的卷积来获取分割预测图,该预测图随后被上采样到与原图像一样的大小,以得到最终的语义分割结果.

深监督.深监督对很多计算机视觉任务都有帮助,例如图像分类[55]、物体检测[56]、视觉跟踪[57]和边缘检测[58]等.本文所提出的MiniNet也采用了深监督来提高性能.在解码路径中,如F_1^D一样,本文将一个1×1卷积和一个上采样操作分别连接在F_2^D、F_3^D和F_4^D之后.训练过程中,所有的这些预测结果都使用真值和标准的softmax损失函数进行监督.与之前的研究[1,3,13]中一样,F_1^D对应的损失函数的权重被设置为1.0,而F_2^D、F_3^D、F_4^D对应的辅助的损失函数的权重被设置为0.4.在测试阶段,F_2^D、F_3^D和F_4^D的预测结果被直接丢弃,并将F_1^D的预测结果作为最终输出的语义分割结果.

模型分析.对于具有r个分支和C个输入、输出通道的SPC模块,其逐点卷积具有C^2个参数,而其金字塔的扩张的深度可分离卷积共有rCt^2个参数.由于卷积核大小$t\times t$实际上是3×3,并且为了效率,令$r\leq 4$,所以逐点卷积占了网络参数的绝大部分.因此,网络的参数量与通道数的平方大约成正比.具有C个通道的SPC模块的参数量大约等于4个具有$C/2$通道的SPC模块的参数量.因此,在网络通道数和网络深度之间取得良好的平衡十分重要.此外,除了多尺度学习之外,MiniNet还容易获得较大的感受野,因为(1)每个SPC模块都具有大的感受野,例如一个具有四分支的SPC模块的感受野为17;(2)大多数MiniNet的网络层在原始图像的1/16尺度上操作.考虑到以上几点,MiniNet的目标是仅使用少量网络参数便能进行精确的语义图像分割.

一些网络设置.本文在表1中设计了两种具体的网络配置.这两种MiniNet变体分别仅有95 K和211 K的网络参数量.对于编码网络中第二、第三和第四个阶段的SPC模块,本文设置SPC分支数(r)分别为3、4、4.更多的分支能够获得更高的精度,但是速度会有所下降.注意,编码网络的第一阶段使用式(1)所示的残差卷积模块而不是SPC模块.

表1　两种具体的MiniNet配置

编号	(C_1,C_2,C_3,C_4)	(P_1,P_2,P_3,P_4)	FLOPs/G	参数量/K
#1	(16,32,64,64)	(2,2,3,4)	2.1	95
#2	(16,32,64,96)	(2,2,3,10)	2.4	211

4　实　验

4.1　实验设置

数据集.本文在三个著名的语义分割数据集上评估了所提出的方法,其中包括Cityscapes数据集[19]、CamVid数据集[27]和Mapillary Vistas数据集[28].Cityscapes数据集[19]包含在50个不同城市的街道场景中记录的各种图像集,它由2975张训练图像、500张验证图像和1525张测试图像以及相应的像素级别的标注组成.所有图像均具有1024×2048的高分辨率,共19个类别,被分为7组.本文在验证集上进行消融实验,并在测试集上与其他方法进行比较.CamVid数据集[27]也是用于城市场景理解的,它包括367张训练图像、101张验证图像和233张测试图像,共有11个类别.所有图像的分辨率均为360×480.本文遵循先前的工作[9,12-13,35]采用训练集和验证集进行训练,并采用测试集进行测试.另外,本文使用Mapillary Vistas数据集[28]来研究网络的泛化性.本文将其验证集中的66个类(2000张图像)映射到Cityscapes数据集中的19个类.随后,使用在Cityscapes数据集上预先训练的模型在该数据集上评测.

实现细节.本文使用流行的PyTorch框架[59]来实现MiniNet.采用Adam[60]优化器来训练,权重衰减系数为1e-4,初始学习率设置为2e-3.本文使用"poly"学习率策略,当前学习率等于基础学习率乘以$(1-curr_iter/max_iter)^{power}$(其中$power=0.9$,$curr_iter$和$max_iter$分别表示当前和总共的迭代次数).当与之前的模型进行比较时,本文遵循之前的研究[9-10]对MiniNet训练300个epoch.但是,当进行消融研究时,本文遵循之前的研究[9],只训练100个epoch以节省时间.本文同样遵循之前的研究[9-10]使用标准的缩放、裁剪和翻转等操作对数据进行增广处理.对于Cityscapes数据集[19],其图像分辨率为1024×2048.本文遵循之前的研究[9-10]将

其图像降采样为 512×1024 以进行训练;而为了进行准确的性能评估,本文使用双线性插值对网络的输出进行上采样,使其变为原始图像的大小,即 1024×2048. 乘法-加法操作(Multiplication-Add Operations,FLOPs)的计算量和推理速度都是在 512×1024 的图像分辨率下进行计算的,该分辨率已经能够满足大多数实际应用的需求. 在 Cityscapes 数据集上训练时,仅使用该数据集中提供的精细标签. MiniNet 是从随机初始化开始训练的,而没有像其他模型一样在 ImageNet 数据集[26]上进行预训练. 在模型测试时,本文直接将 MiniNet 网络的输出作为最终结果,而不使用任何额外的后处理. 本文中所有的实验都是在一块 NVIDIA TITAN Xp GPU 上运行的.

4.2 消融实验

在与以前的分割模型进行比较之前,本文首先评估 MiniNet 中各种设计选择的合理性. 本文使用具有挑战性的 Cityscapes 验证集[19]进行消融实验. 所有的消融实验都将以 MiniNet 的 211K 版本(表 1 中的第二个变体)作为默认设置,在训练集上进行训练并在验证集上进行评测.

下采样大小. MiniNet 将大多数网络层置于 1/16 尺度下,而不是之前常用的 1/8 尺度下[9,11-12,15-17]. 表 2(a)展示了具有不同下采样尺度的实验结果. 从实验结果中可见,与 1/8 尺度相比,MiniNet 的尺度为 1/16 时,精度更高、速度更快且 FLOPs 更少. 当把特征图缩放到更小的尺度时,卷积神经网络的响应野会更大,更能捕获全局的信息,从而使得 1/16 尺度下的精度比 1/8 尺度高. 如前文所述,当把卷积运算放在更小的尺度上时,需要操作的像素点数量会显著减少,因而使得 1/16 尺度下的速度更快且 FLOPs 更少.

长/短残差连接. 从表 2(b)中可见,短残差连接对于 MiniNet 是必不可少的,这再一次验证了残差网络[24]的有效性. 并且长残差连接能够进一步提高网络的效果,同时并不会显著增加计算量. 以后的研究也可以仿照本文在同一个尺度的卷积之间添加长残差连接.

SPP 模块的两个分支. SPP 模块具有两个分支,在表 2(c)中评估了这种设计的有效性. 注意,仅有池化分支的 MiniNet 在编码器的第一阶段中使用带有步长的卷积来进行下采样,否则的话,将直接对图像用池化进行下采样. 从表 2(c)中可见,带有两个分支的 SPP 模块效果最佳,这说明在轻量级语义分割中设计两个分支的 SPP 模块来进行下采样是十分必要的,可以减少下采样的信息损失.

深监督. 如表 2(d)所示,采用深监督来提高效果是十分必要的. 由于在网络推理时,深监督相关的计算会被丢掉,所以深监督可以在没有任何消耗的情况下将 mIoU 从 64.8%提高到 65.5%. 而在之前

表 2 MiniNet 的消融实验的结果

(a) 不同下采样尺度的效果

尺度	参数量/K	FLOPs/G	速度/fps	mIoU
1/8	217	3.3	64.9	64.9
1/16	**211**	**2.4**	**94.3**	**65.5**

(b) 所提出的长/短残差连接的影响

残差连接	参数量/K	FLOPs/G	速度/fps	mIoU
w/o long	196	2.3	98.0	64.8
w/o short	211	2.4	96.2	62.7
w/all	**211**	**2.4**	**94.3**	**65.5**

(c) SPP 模块中每个分支的影响

分支	参数量/K	FLOPs/G	速度/fps	mIoU
pool.	194	2.2	101.0	64.0
conv.	219	2.5	91.7	64.6
both	**211**	**2.4**	**94.3**	**65.5**

(d) 深监督的影响

深监督	参数量/K	FLOPs/G	速度/fps	mIoU
w/o	211	2.4	94.3	64.8
w/	**211**	**2.4**	**94.3**	**65.5**

(e) MiniNet 解码器的影响

解码器	参数量/K	FLOPs/G	速度/fps	mIoU
w/o	182	1.5	156.3	61.4
w/	**211**	**2.4**	**94.3**	**65.5**

(f) PReLU 激活函数的影响

激活函数	参数量/K	FLOPs/G	速度/fbs	mIoU
ReLU	209	2.4	94.3fps	64.5
PReLU	**211**	**2.4**	**94.3fps**	**65.5**

(g) 不同卷积通道数量的效果

(C_1,C_2,C_3,C_4)	参数量/K	FLOPs/G	速度/fps	mIoU
(16,32,32,64)	108	1.7	106.4	62.6
(16,32,64,64)	135	2.2	98.0	63.6
(16,32,64,96)	**211**	**2.4**	**94.3**	**65.5**
(16,32,64,128)	318	2.8	88.5	65.7
(32,64,96,128)	376	5.6	68.5	67.3
(16,64,128,128)	415	4.0	67.6	67.6
(32,64,128,128)	431	6.1	65.4	67.4

(h) 不同数量的 SPC 模块的效果

(P_1,P_2,P_3,P_4)	参数量/K	FLOPs/G	速度/fps	mIoU
(1,1,3,10)	207	2.0	101.0	65.0
(2,2,3,10)	**211**	**2.4**	**94.3**	**65.5**
(2,2,5,10)	224	2.5	87.0	66.3
(3,3,5,10)	229	2.9	80.0	65.8
(2,2,3,5)	146	2.3	99.0	63.3
(2,2,3,8)	185	2.4	96.2	64.5
(2,2,3,12)	237	2.5	91.7	65.2

关于语义分割的研究中，往往忽略了深监督对语义分割的作用.

解码器. 表 2(e)表明 MiniNet 中的解码器在将特征图从一个很小的分辨率(即 1/16)解码到原图大小中扮演了十分重要的角色. 移除解码器，直接上采样 1/16 尺度下的预测作为输出结果，将非常显著地降低网络性能. 这样的实验结果是比较直观的，因为直接从 1/16 尺度上采样到原图大小将会损失很多细节信息.

PReLU 非线性激活函数. 本文遵循 ESPNet[9] 采用 PReLU[51] 作为非线性激活函数，而不是最常用的 ReLU[63] 激活函数. 为了验证这一选择的有效性，本文使用 ReLU 非线性激活函数代替了 PReLU 非线性激活函数. 结果显示在表 2(f)中. PReLU 可以将 ReLU 的 mIoU 从 64.5%提高到 65.5%. 因此，PReLU 激活函数在轻量级分割任务上优于 ReLU 激活函数.

卷积通道数. 在表 2(g)中评估了不同卷积通道数的影响. 更多的通道数能够产生更好的结果，但同时伴随着更多的参数、更多的 FLOPs 以及更低的速度. 考虑到精度、速度、参数量等之间的权衡，对于 MiniNet 的两种变体，本文选择了两组(C_1，C_2，C_3，C_4)的设置，即(16，32，64，64)和(16，32，64，96).

SPC 模块的数量. 表 2(h)展示了不同 SPC 模块数量的评估结果. 添加更多的 SPC 模块能够产生更好的结果，对于编码过程的第三个阶段(即 P_3)尤其有效，因为当前 P_3 较小. 考虑到精度、速度、参数量等之间的权衡，本文设置(2，2，3，10)作为 211K 参数版本的 MiniNet 的设置.

4.3　与最优模型的比较

Cityscapes 数据集. 本小节将本文所提出的 MiniNet 在 Cityscapes 数据集上[19]与高精度语义分割模型以及轻量级模型进行比较，包括 DANet[61]、DeepLabv3＋[4]、BiSeNet[14]、DenseASPP[40]、DFN[5]、PSPNet[1]、FRRN[48]、SegNet[35]、DeepLabv1[38]、DeepLabv2[23]、FCN-8s[21]、ICNet[13]、ENet[12]、ShuffleNetv2[62]、ESPNet[9] 和 ESPNetv2[10]. 本文报告这些模型的参数数量、FLOPs 和速度. 对于 Cityscapes 验证集的精度评测采用所有类别的 IoU 的均值，对于测试集同时采用所有类别的 IoU 的均值和 7 大类别分组的 IoU 的均值. 结果如表 3 所示.

表 3　在 Cityscapes 数据集[19]上 MiniNet 与其他分割模型的对比("—"表明本文无法获取相应的结果)

方法	预训练	参数量	FLOPs	速度/fps	mIoU/%		
					class (*val*)	category (*test*)	class (*test*)
DANet[61]	ImageNet	68.50M	551.7G	<1	81.5	—	81.5
DeepLabv3＋[4]	**ImageNet＋COCO＋JFT**	**54.61M**	**165.9G**	**<1**	**79.6**	**—**	**82.1**
BiSeNet[14]	ImageNet	5.80M	6.6G	42.0	69.0	—	68.4
DenseASPP[40]	**ImageNet**	**28.64M**	**244.9G**	**<1**	**78.9**	**90.7**	**80.6**
DFN[5]	ImageNet＋COCO	44.84M	165.9G	1.2	—	—	79.3
PSPNet[1]	**ImageNet＋COCO**	**65.58M**	**514.0G**	**<1**	**—**	**90.6**	**80.2**
FRRN[48]	ImageNet	17.71M	475.8G	5.0	—	88.9	71.8
SegNet[35]	**ImageNet**	**29.45M**	**326.0G**	**8.0**	**—**	**79.1**	**57.0**
DeepLabv1[38]	ImageNet	42.54M	362.9G	1.0	—	—	63.1
DeepLabv2[23]	**ImageNet＋COCO**	**43.90M**	**374.3G**	**<1**	**71.4**	**86.4**	**70.4**
FCN-8s[21]	ImageNet	134.46M	334.4G	6.8	—	85.7	65.3
ICNet[13]	**ImageNet**	**6.70M**	**7.4G**	**62.5**	**67.7**	**—**	**69.5**
ShuffleNetv2[62]	ImageNet	2.60M	3.5G	91.7	60.3	—	—
ENet[12]	**No**	**364K**	**3.8G**	**34.7**	**—**	**80.4**	**58.3**
ESPNet[9]	No	364K	4.5G	61.0	61.4	82.2	60.3
ESPNetv2[10]	**ImageNet**	**99K**	**564M**	**142.0**	**54.1**	**—**	**54.7**
ESPNetv2[10]	ImageNet	725K	3.4G	83.0	62.7	—	62.1
MiniNet	No	95K	2.1G	104.2	63.3	84.2	64.1
MiniNet	No	211K	2.4G	94.3	67.3	85.1	66.3

首先来分析所提出的 MiniNet 与高精度模型的比较. MiniNet 具有最少的参数量，即便较精确版本的 MiniNet 也只有 211K 参数，而 DANet[61] 的参数量是 MiniNet 的 320 倍，DeepLabv3＋[4] 的参数量是 MiniNet 的 250 倍. MiniNet 的少量参数使其可以灵活地被部署到各种移动设备上，精确版的 MiniNet 仅需要不到 1M 的存储空间，而 95K 参数版本的 MiniNet 仅需要 0.4M 的存储空间. 此外，精确版的 MiniNet 仅有 2.4G FLOPs，比其他高精度模型少两个数量级. 很少的 FLOPs 意味着 MiniNet 的能耗很小，使其适合在移动设备上的部署. 精确版的 MiniNet 还实现了 94.3fps 的超实时速度，而大

多数高精度模型甚至达不到 1fps 的速度. 与 FCN-8s[21]和 DeepLabv1[38]相比,MiniNet 具有更快的速度、更少的参数、更少的 FLOPs、更高的准确性,且 MiniNet 的精度可与 DeepLabv2[23]媲美. 另外,MiniNet 不需要像很多大型网络一样在 ImageNet 数据集[26]上预训练. 这是因为从随机初始化开始训练小型网络比较容易,而没有在 ImageNet 数据集上预训练的大型网络则很难收敛. 没有任何预训练使得开发新的网络结构十分灵活. 例如,如第 4.2 节所示,用户可以轻松地通过堆叠适当更多的模块或适当增大卷积通道数来获得更好的性能,而无需进行预训练.

接着来分析所提出的 MiniNet 与高精度模型的比较. 从表 3 中可见,ESPNetv2[10]具有比 ESPNet[9]更好的精度,但是 ESPNetv2 参数更多. 然而,具有 95K 参数的 MiniNet 却比具有 725K 参数的 ESPNetv2[10]具有更好的效果. 具体来说,ESPNetv2[10]的参数是 95K 版本 MiniNet 的 7.6 倍,且 MiniNet 的 FLOPs 更少、速度更快、准确性更高. 与 ESPNetv2 的 99K 版本相比,MiniNet 的 95K 版本的 mIoU 高出 9% 以上. 与 ICNet[13]相比,ICNet 的参数量是 MiniNet 的 30 倍,且 MiniNet 的速度更快,性能也能与其媲美. MiniNet 也大大优于 ShuffleNetv2[62]. 请注意,MiniNet 并未使用 ImageNet[26]预训练,而大多数其他网络都已在 ImageNet 上进行了预训练. 这些实验结果表明,MiniNet 通过设计 SPP 和 SPC 模块实现了更有效的轻量级多尺度特征学习,并通过控制下采样尺度、神经网络深度与通道数的关系,取得了语义分割精度、效率、参数及计算量之间的良好平衡.

图 3 展示了 MiniNet 及其他方法的参数量与在 Cityscapes 测试集[19]上的类别 mIoU 的关系图. 从该图可见,本文的网络可以用最少的参数获得可观的性能. 此外,在图 4 中展示了一些 MiniNet 与最近最流行的轻量级语义分割模型 ESPNetv2[10]的可视化的对比结果.

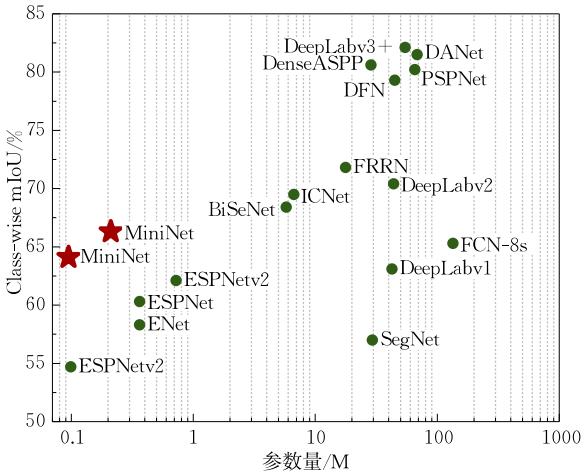


图 3 各种语义分割方法的网络参数量与在 Cityscapes 测试集[19]上的逐类别 mIoU 的关系(左上的点表示模型的精度越高、参数越少)

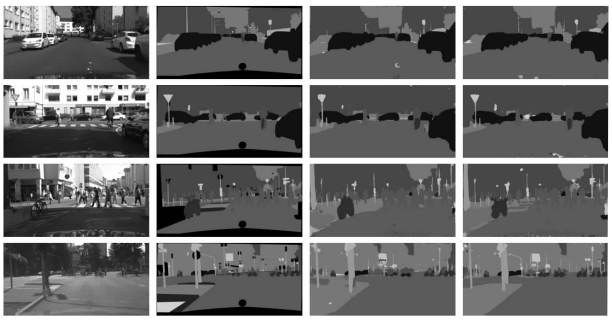


图 4 所提出的 MiniNet 在 Cityscapes 数据集[19]上的分割结果与 ESPNetv2[10]的对比

CamVid 数据集. 接下来在另一个城市场景理解数据集(即 CamVid 数据集[27])上评测所提出的网络. 本文与可获取该数据集上结果的最新分割模型进行了比较,包括 BiSeNet[14]、PSPNet50[1]、SegNet[35]、Dilation8[18]、DeepLabv1[38]、FCN-8s[21]、ICNet[13]、ENet[12]和 ESPNet[9]. 结果如表 4 所示. 由于 CamVid 含有 11 个类别,与 Cityscapes(19 类)有所差别,因此每个分割模型的参数数量与表 3 可能略有不同. 可以发现 MiniNet 仅需少量参数和少量的 FLOPs 即可达到最优的精度,例如,ICNet[13]的参数量是 MiniNet 的 30 倍,PSPNet50[1]的参数量是 MiniNet 的 220 倍.

表 4 所提出的 MiniNet 与其他分割模型在 CamVid 测试数据集[27]上的效果比较

方法	参数量	FLOPs/G	mIoU/%
BiSeNet[14]	5.80M	2.2	65.6
PSPNet50[1]	46.58M	117.2	69.1
SegNet[35]	29.45M	104.3	55.6
Dilation8[18]	140.8M	—	65.3
DeepLabv1[38]	42.52M	121.4	61.6
FCN-8s[21]	134.35M	139.6	57.0
ICNet[13]	6.68M	2.6	67.1
ENet[12]	364K	1.3	51.3
ESPNet[9]	353K	1.3	55.6
MiniNet	95K	0.7	66.7
MiniNet	211K	0.8	67.5

Mapillary Vistas 数据集. Mapillary Vistas 数据集[28]是最新发布的街道场景数据集. 本文通过如下方式将 Mapillary Vistas 数据集中的 66 个类别映射到 Cityscapes 数据集[19]中的 19 个类别:(1)合并“traffic sign front”类和“traffic sign back”类到

Cityscapes 中的“traffic sign”类；(2) 合并“bicyclist”、“motorcyclist”和“other rider”到 Cityscapes 中的“rider”类；(3) 忽略 Mapillary Vistas 没有出现在 Cityscapes 中的其他类. 因此，得到的新 Mapillary 数据集将具有与 Cityscapes 数据集相同的类别. 为了测试分割模型对未见数据的泛化性，本文使用在 Cityscapes 训练集[19]上预训练的模型来对 Mapillary 验证集(2000 张图像)进行评估，模型并不进行任何微调.

在 ESPNet[9]中，Mehta 等人提议将 Mapillary 的类别划分为与 Cityscapes 相同的 7 个类别分组. 这是不适当的，因为这样的分类将使得看起来完全不一样的物体被分到同一个分组中. 例如，“vehicle”分组包括了“bus”和“car”等类别，但是 Mapillary 中的“boat”类别也被看作了“vehicle”. 对于在“bus”和“car”类别的数据上预训练的模型，是很难检测出“boat”的，因为它们看起来完全不同.

结果总结在表 5 中. 尽管 MiniNet 仍然比大规模、参数更多的网络(例如 PSPNet[1]和 ICNet[13])差一些，但它始终优于其他轻量级语义分割模型，包括 ENet[12]、ESPNet[9]和 ESPNetv2[10].

表 5　Mapillary Vistas 验证集[28]上的模型泛化性评估(该数据集中的类别已被映射到与 Cityscapes 数据集的类别相同，所有的模型均在 Cityscapes 训练集上训练并且没有再微调)

方法	参数量	Pixel Acc. /%	mIoU/%
PSPNet[1]	65.58M	82.5	43.8
ICNet[13]	6.68M	78.1	31.8
ENet[12]	364K	50.1	18.0
ESPNet[9]	364K	50.5	16.2
ESPNetv2[10]	99K	47.0	13.2
ESPNetv2[10]	725K	43.4	14.6
MiniNet	95K	55.7	18.2
MiniNet	211K	55.8	20.5

5　总　结

本文提出了一种新的轻量级语义分割网络MiniNet，该网络通过 SPC 和 SPP 模块实现了多尺度学习，将大多数网络层置于较小的分辨率(1/16 尺度)，并试图平衡卷积通道数和网络深度. 本文提供了详细的消融实验，以证明各种设计选择的有效性，这将有助于对 MiniNet 的理解. 与最新的分割模型相比，MiniNet 能够以更少的参数、更快的速度和更少的 FLOPs 获得更好或相当的准确性. MiniNet 的高效率和小尺寸使其可以部署在移动设备上. 将来，我们计划将 MiniNet 应用于其他移动端的视觉任务.

参　考　文　献

[1] Zhao H, Shi J, Qi X, et al. Pyramid scene parsing network//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017: 2881-2890

[2] Lin G, Milan A, Shen C, et al. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017: 1925-1934

[3] Zhang H, Dana K, Shi J, et al. Context encoding for semantic segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 7151-7160

[4] Chen L C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation//Proceedings of the European Conference on Computer Vision. Munich, Germany, 2018: 801-818

[5] Yu C, Wang J, Peng C, et al. Learning a discriminative feature network for semantic segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 1857-1866

[6] Cermelli F, Mancini M, Bulo S R, et al. Modeling the background for incremental learning in semantic segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2020: 9233-9242

[7] Siddiqui Y, Valentin J, Niessner M. ViewAL: Active learning with viewpoint entropy for semantic segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2020: 9433-9443

[8] Zhang Y, Qiu Z, Yao T, et al. Transferring and regularizing prediction for semantic segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2020: 9621-9630

[9] Mehta S, Rastegari M, Caspi A, et al. ESPNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation//Proceedings of the European Conference on Computer Vision. Munich, Germany, 2018: 552-568

[10] Mehta S, Rastegari M, Shapiro L, et al. ESPNetv2: A light-weight, power efficient, and general purpose convolutional neural network//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, USA, 2019: 9190-9200

[11] Romera E, Alvarez J M, Bergasa L M, et al. ERFNet: Efficient residual factorized ConvNet for real-time semantic segmentation. IEEE Transactions on Intelligent Transportation Systems, 2018, 19(1): 263-272

[12] Paszke A, Chaurasia A, Kim S, et al. ENet: A deep neural network architecture for real-time semantic segmentation. arXiv preprint arXiv:1606.02147, 2016

[13] Zhao H, Qi X, Shen X, et al. ICNet for real-time semantic segmentation on high-resolution images//Proceedings of the European Conference on Computer Vision. Munich, Germany, 2018: 405-420

[14] Yu C, Wang J, Peng C, et al. BiSeNet: Bilateral segmentation network for real-time semantic segmentation//Proceedings of the European Conference on Computer Vision. Munich, Germany, 2018: 325-341

[15] Poudel R P, Bonde U, Liwicki S, et al. ContextNet: Exploring context and detail for semantic segmentation in real-time//Proceedings of the British Machine Vision Conference. Newcastle upon Tyne, UK, 2018: 1-11

[16] Lo S Y, Hang H M, Chan S W, et al. Efficient dense modules of asymmetric convolution for real-time semantic segmentation//Proceedings of the ACM Multimedia Asia. Beijing, China, 2019: 1-6

[17] Wu T, Tang S, Zhang R, et al. CGNet: A light-weight context guided network for semantic segmentation. IEEE Transactions on Image Processing, 2020, 30: 1169-1179

[18] Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions//Proceedings of the International Conference on Learning Representations. San Juan, Puerto Rico, 2016: 1-13

[19] Cordts M, Omran M, Ramos S, et al. The Cityscapes dataset for semantic urban scene understanding//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 3213-3223

[20] Hariharan B, Arbeláez P, Girshick R, et al. Hypercolumns for object segmentation and fine-grained localization//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA, 2015: 447-456

[21] Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(4): 640-651

[22] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional networks for biomedical image segmentation//Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention. Munich, Germany, 2015: 234-241

[23] Chen L C, Papandreou G, Kokkinos I, et al. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4): 834-848

[24] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 770-778

[25] Yu F, Wang D, Shelhamer E, et al. Deep layer aggregation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 2403-2412

[26] Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge. International Journal of Computer Vision, 2015, 115(3): 211-252

[27] Brostow G J, Shotton J, Fauqueur J, et al. Segmentation and recognition using structure from motion point clouds//Proceedings of the European Conference on Computer Vision. Marseille, France, 2008: 44-57

[28] Neuhold G, Ollmann T, Rota Bulo S, et al. The Mapillary Vistas dataset for semantic understanding of street scenes//Proceedings of the International Conference on Computer Vision. Venice, Italy, 2017: 4990-4999

[29] Luo Hui-Lan, Lu Fei, Kong Fan-Sheng. Image semantic segmentation based on region and deep residual network. Journal of Electronics & Information Technology, 2019, 41(11): 2777-2786(in Chinese)
(罗会兰, 卢飞, 孔繁胜. 基于区域与深度残差网络的图像语义分割. 电子与信息学报, 2019, 41(11): 2777-2786)

[30] Zhang Shun, Gong Yi-Hong, Wang Jin-Jun. The development of deep convolution neural network and its applications on computer vision. Chinese Journal of Computers, 2019, 42(3): 453-482(in Chinese)
(张顺, 龚怡宏, 王进军. 深度卷积神经网络的发展及其在计算机视觉领域的应用. 计算机学报, 2019, 42(3): 453-482)

[31] Wei Yun-Chao, Zhao Yao. A review on image semantic segmentation based on DCNN. Journal of Beijing Jiaotong University, 2016, 40(4): 82-91(in Chinese)
(魏云超, 赵耀. 基于 DCNN 的图像语义分割综述. 北京交通大学学报, 2016, 40(4): 82-91)

[32] Jiang Feng, Gu Qing, Hao Hui-Zhen, et al. Survey on content-based image segmentation methods. Journal of Software, 2017, 28(1): 160-183(in Chinese)
(姜枫, 顾庆, 郝慧珍等. 基于内容的图像分割方法综述. 软件学报, 2017, 28(1): 160-183)

[33] Zhao Fei, Zhang Wen-Kai, Yan Zhi-Yuan, et al. Multi-feature map pyramid fusion deep network for semantic segmentation on remote sensing data. Journal of Electronics & Information Technology, 2019, 41(10): 2525-2531(in Chinese)
(赵斐, 张文凯, 闫志远等. 基于多特征图金字塔融合深度网络的遥感图像语义分割. 电子与信息学报, 2019, 41(10): 2525-2531)

[34] Noh H, Hong S, Han B. Learning deconvolution network for semantic segmentation//Proceedings of the International Conference on Computer Vision. Santiago, Chile, 2015: 1520-1528

[35] Badrinarayanan V, Kendall A, Cipolla R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495

[36] Chen L C, Yang Y, Wang J, et al. Attention to scale: Scale-aware semantic image segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 3640-3649

[37] Xia F, Wang P, Chen L C, et al. Zoom better to see clearer: Human and object parsing with hierarchical auto-zoom net//Proceedings of the European Conference on Computer Vision. Amsterdam, The Netherlands, 2016: 648-663

[38] Chen L C, Papandreou G, Kokkinos I, et al. Semantic image segmentation with deep convolutional nets and fully connected CRFs//Proceedings of the International Conference on Learning Representations. San Diego, USA, 2015: 1-14

[39] Chen L C, Papandreou G, Schroff F, et al. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587, 2017

[40] Yang M, Yu K, Zhang C, et al. DenseASPP for semantic segmentation in street scenes//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 3684-3692

[41] Huang Z, Wang X, Huang L, et al. CCNet: Criss-cross attention for semantic segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, USA, 2019: 603-612

[42] Zhu Z, Xu M, Bai S, et al. Asymmetric non-local neural networks for semantic segmentation//Proceedings of the International Conference on Computer Vision. Seoul, Korea, 2019: 593-602

[43] Wu Z, Shen C, Hengel A V D. Wider or deeper: Revisiting the ResNet model for visual recognition. Pattern Recognition, 2019, 90: 119-133

[44] Liu Z, Li X, Luo P, et al. Semantic image segmentation via deep parsing network//Proceedings of the International Conference on Computer Vision. Santiago, Chile, 2015: 1377-1385

[45] Zheng S, Jayasumana S, Romera-Paredes B, et al. Conditional random fields as recurrent neural networks//Proceedings of the International Conference on Computer Vision. Santiago, Chile, 2015: 1529-1537

[46] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA, 2015: 1-9

[47] Treml M, Arjona-Medina J, et al. Speeding up semantic segmentation for autonomous driving//Proceedings of the Workshop of Machine Learning for Intelligent Transportation Systems, Annual Conference on Neural Information Processing Systems. Barcelona, Spain, 2016: 1-7

[48] Pohlen T, Hermans A, Mathias M, et al. Full-resolution residual networks for semantic segmentation in street scenes//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017: 4151-4160

[49] Howard A G, Zhu M, Chen B, et al. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017

[50] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift//Proceedings of the International Conference on Machine Learning. Lille, France, 2015: 448-456

[51] He K, Zhang X, Ren S, et al. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification //Proceedings of the International Conference on Computer Vision. Santiago, Chile, 2015: 1026-1034

[52] Wang X, Girshick R, Gupta A, et al. Non-local neural networks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 7794-7803

[53] Chollet F. Xception: Deep learning with depthwise separable convolutions//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017: 1251-1258

[54] Sandler M, Howard A, Zhu M, et al. MobileNetV2: Inverted residuals and linear bottlenecks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 4510-4520

[55] Lee C Y, Xie S, Gallagher P, et al. Deeply-supervised nets//Proceedings of the International Conference on Artificial Intelligence and Statistics. San Diego, USA, 2015: 562-570

[56] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 42(2): 318-327

[57] Wang L, Ouyang W, Wang X, et al. Visual tracking with fully convolutional networks//Proceedings of the International Conference on Computer Vision. Santiago, Chile, 2015: 3119-3127

[58] Liu Y, Cheng M M, Hu X, et al. Richer convolutional features for edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(8): 1939-1946

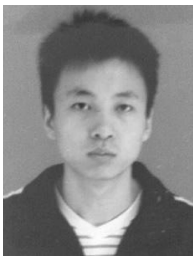
[59] Paszke A, Gross S, Massa F, et al. PyTorch: An imperative style, high-performance deep learning library//Proceedings of the Annual Conference on Neural Information Processing Systems. Vancouver, Canada, 2019: 8026-8037

[60] Kingma D P, Ba J. Adam: A method for stochastic optimization //Proceedings of the International Conference on Learning Representations. San Diego, USA, 2015: 1-15

[61] Fu J, Liu J, Tian H, et al. Dual attention network for scene segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, USA, 2019: 3146-3154

[62] Ma N, Zhang X, Zheng H T, et al. ShuffleNet v2: Practical guidelines for efficient CNN architecture design//Proceedings of the European Conference on Computer Vision. Munich, Germany, 2018: 116-131

[63] Nair V, Hinton G E. Rectified linear units improve restricted Boltzmann machines//Proceedings of the International Conference on Machine Learning. Haifa, Israel, 2010: 807-814



LIU Yun, Ph. D. candidate. His research interest is computer vision.

LU Cheng-Ze, M. S. candidate. His research interest is computer vision.

LI Shi-Jie, Ph. D. candidate. His research interest is computer vision.

ZHANG Le, Ph. D., research scientist. His research interest is deep learning.

WU Yu-Huan, Ph. D. candidate. His research interest is computer vision.

CHENG Ming-Ming, Ph. D., professor. His research interests include computer vision and computer graphics.

Background

Semantic segmentation is a fundamental problem in computer vision, which aims at assigning a semantic label for each pixel in an image. It is common for recent convolutional neural networks (CNN) to improve the accuracy by introducing more parameters and operations. However, mobile devices cannot deploy powerful GPUs that are large and power-hungry, so that the limited computational resources prevent the application of state-of-the-art segmentation models. In addition, mobile devices also have restrictive storage space. For example, it is impossible for smartphones to use hundreds of MB of memory to store a pretrained deep model for a specific application. This inspires us to develop semantic segmentation models with good trade-offs among accuracy, efficiency, the number of parameters and power consumption.

To this end, recent interest in lightweight semantic segmentation is rapidly increasing. Many lightweight segmentation models have emerged. These models usually adopt depthwise separable convolutions, asymmetric convolutions, and dense connections to reduce the network parameters and operations. Although the state of the arts has been pushed to some extent, current models suffer from either unsatisfactory accuracy or too many parameters. For example, ESPNetv2 has 0.73M parameters but only achieves 62.1% mIoU on the Cityscapes test dataset, while ICNet achieves 69.5% mIoU but has 6.68M parameters (i. e., $6.68\text{M} \times 4 = 26.72\text{M}$ storage memory). To enable a model to be flexibly deployed on mobile devices, we believe the number of parameters should be less than 0.5M, with less than $0.5\text{M} \times 4 = 2.0\text{M}$ storage memory.

In this paper, we propose a lightweight, fast and power-efficient semantic segmentation network, namely MiniNet, for mobile applications. In our design, we introduce simple yet effective spatial pyramid convolution (SPC) and spatial pyramid pooling (SPP) modules as the basic units of MiniNet to learn multi-scale representations from natural images. To accelerate the computation of MiniNet, we downsample the feature map into 1/16 of the original resolution and put most layers and operations at this small scale, unlike previous lightweight models that put most layers at the 1/8 scale. The small feature scale also helps MiniNet enlarge the receptive fields, so that MiniNet can learn high-level abstract semantic information with fewer parameters. We also manage to make a good balance between network depth and the number of convolution channels for better performance. Due to the small model size, MiniNet does not need ImageNet pretraining, which makes it flexible to adapt to new data and new tasks. On the Cityscapes test dataset, without ImageNet pretraining, MiniNet achieves 66.3% mIoU with 211K parameters and 2.4G FLOPs, running at 94.3fps. A smaller version of MiniNet with 95K parameters achieves 64.1% mIoU with a speed of 104.2fps.

This work was supported by the Major Projects of New Generation Artificial Intelligence (2018AAA0100400), the National Natural Science Foundation for Outstanding Young Scholars (61922046), and the Scientific Innovation Planning Projects Guided by the Ministry of Education.