一种基于改进的 MobileNetV2 网络语义分割算法

孟 琭[1],徐 磊[1],郭嘉阳[2]
(1. 东北大学信息科学与工程学院,辽宁沈阳 110000; 2. 辛辛那提大学电气工程与计算机系,俄亥俄州辛辛那提 45221)

摘 要: 基于金字塔卷积神经网络的语义分割算法准确率很高,但是其计算资源消耗巨大、算法执行时间长、无法满足实时性要求.为了解决这个问题,本文做出了以下改进:(1)用 MobileNet 替换原网络的结构,减少了网络运算时间和内存开销;(2)引入编码器-解码器结构提高输出图像的分辨率,进一步细化分割结果;(3)针对高分辨率图像推断时间过长的问题,本文设计了多级图像输入方法,降低了网络推断高分辨率图像所消耗的时间.本文在 VOC 2012 数据集和 Cityscapes 数据集上进行了测试,并与 FCN、SegNet、DeepLab、PSPNet 以及 DFN 等语义分割模型对比.实验结果表明,本文设计的语义分割算法在 VOC 2012 数据集上达到了 76.1% 的 mIoU,在 Cityscapes 数据集上达到了 74.1% 的 mIoU,略低于传统语义分割算法;处理一张分辨率为 1024×512 的图片需要 18ms,少于传统语义分割算法,满足了实时性要求,达到了准确率与计算资源消耗之间的平衡.

关键词: 语义分割; 卷积神经网络; 金字塔网络; 快速语义分割; MobileNet; 编码器-解码器
中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112 (2020)09-1769-08
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2020.09.015

Semantic Segmentation Algorithm Based on Improved MobileNetV2

MENG Lu[1], XU Lei[1], GUO Jia-yang[2]
(1. *College of Information Science and Engineering, Northeastern University, Shenyang, Liaoning* 110000, *China*;
2. *Department of Electrical Engineering and Computer Science, University of Cincinnati, Cincinnati, Ohio* 45221, *USA*)

Abstract: The algorithm of semantic segmentation based on pyramid convolution neural network has high accuracy, but it consumes a lot of computing resources, takes a long time to execute, and cannot meet the real-time requirements. To overcome these shortcomings, this paper made the following improvements: (1) replacing the original network structure with MobileNet in order to reduce the computation time and memory consumption; (2) using encoder-decoder structure to improve the resolution of the output image and further refine the segmentation results; (3) using a multi-level image input method, which can reduce the time consumed by network inference of high-resolution image. Our method was tested on the VOC 2012 dataset and the Cityscapes dataset compared with other state-of-the-art semantic segmentation models such as FCN(Fully Convolutional Networks), SegNet, DeepLab, PSPNet and DFN(Discriminative Feature Network). Experimental results showed that our method achieved mIoU of 76.1% on the VOC 2012 dataset, and achieved mIOU of 74.1% on the Cityscapes dataset, which was a little lower than the traditional semantic segmentation algorithms. It took 18ms for our method to predict a 1024×512 picture, which achieved a balance between accuracy and computational resource consumption.

Key words: semantic segmentation; convolution neural network; pyramid network; fast semantic segmentation; MobileNet; encoder-decoder

1 引言

语义分割的定义是为图像中的每个像素分配一个事先定义好的语义标签[1,2],语义分割在现实中有许多应用场景[3~7],例如卫星图像处理、自动驾驶、人脸分割、计算机辅助诊断等[8~10].语义分割的实现方法大体可分为三个阶段:

2010年以来,基于深度学习的图像语义分割模型成为了主流,基于深度学习的语义分割算法还可分为两类:一类是基于区域的(Region-based Convolution Neu-

收稿日期:2019-09-10;修回日期:2019-12-27;责任编辑:覃怀银
基金项目:国家自然科学基金(No.61973058);教育部中央高校基本科研基金(No.N2004020)

ral Network,RCNN)[11],另一类是基于全卷积网络(Full Convolution Network,FCN).

文献[12]在2014年率先提出了全卷积网络(FCN).FCN将VGG-Net[13]、AlexNet和GoogleNet[14]中的最后几个全连接层替换为卷积层,使得网络只由卷积层组成,因此被称为全卷积网络. DilatedNet首次引入了空洞卷积来提升网络感受野的大小,同时保持输出大小不变[15]. U-Net同样使用了FCN作为网络的整体结构[16]. 随着深度残差网络(ResNet)在2015年的提出,残差结构迅速地影响了各种CNN模型的设计[17]. DeepLab的V1版本使用VGG作为基本组成结构,而在V2、V3版本则使用ResNet作为基本结构[18]. DeepLabV3中放弃了CRF作为后处理操作,通过空洞卷积和空洞空间金字塔池化(ASPP)已经可以获得比较细致的分割预测结果[19]. PSPNet[20]的作者设计了金字塔池化模块(PPM),通过四种不同尺寸的池化在ResNet的输出特征图上进行操作,之后再将各个分支连接起来,用卷积层做最后的分类. DFN借鉴了SENet[21]中的squeeze-and-excitation操作,设计了细化残差块(Refinement Residual Block,RRB)和通道注意力块(Channel Attention Block,CAB)[22~25].

以上几种基于FCN的语义分割模型虽然在准确率上达到了较高的水平,但在速度上却不尽如人意. 即使使用GPU加速,上述的模型也难以在高分辨率图像上达到实时的处理速度. 所以在计算资源和内存资源受限的条件下难以应用,例如:嵌入式设备方面;在对实时性要求比较高的领域,也不能直接应用,例如:自动驾驶方面.

为了解决这一问题,需要在计算资源消耗、计算速度以及分割准确性之间取得一个平衡,这对于语义分割模型的设计提出了极高的要求,为此本文设计了一种改进的基于FCN的语义分割模型来解决这一难题. 快速语义分割目前是一个研究比较少的领域,如何在尽量少降低准确率的同时大幅度的缩减网络推断所需的时间和空间消耗,是本文所设计的网络主要关注的问题.

2 方法

本文所搭建的低消耗、实时语义分割模型主要由四部分组成,分别是:MobileNetV2、空间金字塔池化(Atrous Spatial Pyramid Pooling,ASPP)、解码器、多级图像输入,如图1所示. 全尺寸输入图像是指原图像,半尺寸输入图像是指将原图的宽、高均缩小为1/2所得到的图像,红色线条表示半尺寸输入图像通过的路径,蓝色线条表示全尺寸输入图像通过的路径,黑色线条表示两者都通过的路径. 半尺寸输入图像穿过整个网络,包括MobileNet、ASPP、解码器、特征融合. 全尺寸的图像与半尺寸图像共享MobileNet的Block1、Block2、Block3、Block4的参数,Block4输出的特征图与半尺寸图像的解码器输出的特征图融合到一起. 最后通过1×1的卷积进行最后的分类,其通道数就是欲分类的个数. 输出的图像尺寸等于全尺寸图像的大小,由全尺寸图像的处理结果和半尺寸图像的处理结果的上采样融合后得到. 2.1~2.4节分别详细介绍本文所提出模型的各个部分.

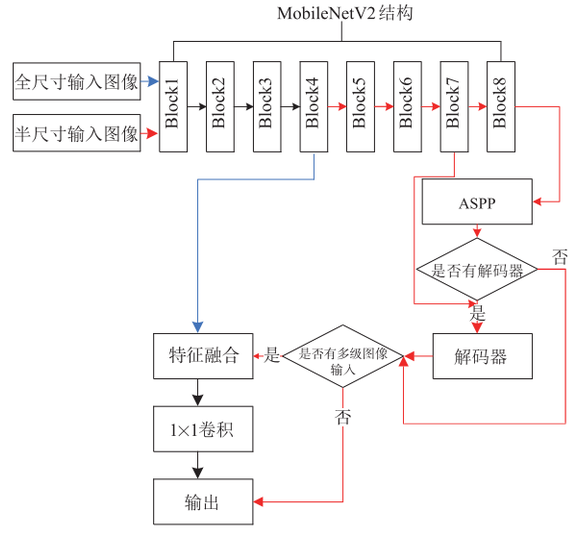


图1 网络的整体结构图,图中红色线条表示半尺寸输入图像通过的路径;蓝色线条表示全尺寸输入图像通过的路径;黑色线条表示两者都通过的路径

2.1 MobileNetV2

MobileNet由Google在2017年提出[26],其主要目的是有效利用移动设备和嵌入式设备的计算资源,提升模型的准确性. MobileNet主要由深度可分离卷积组成的,MobileNetV2使用反向残差结构进一步地改善了网络的性能[27]. 本文选择MobileNetV2是因为该结构相对于普通的FCN减少了8~9倍的计算资源消耗,但分割准确度只是略微下降,比较符合本文方法的低消耗和实时性的要求. 原始的MobileNetV2的网络各层结构如表1所示,通道数代表每层输入的通道数;操作包括普通卷积(conv2d)、反向残差结构的深度可分离卷积(bottleneck)和平均池化(avgpool);OS表示output stride,代表输入图像的大小和输出图像的大小的比值;t表示中间卷积通道的扩张系数;c表示输出通道个数;n表示该层重复了几次;s表示卷积的stride.

本文对原始的MobileNetV2做出了如下改进:(1)为了减少计算开销和内存占用,本文方法仅采用MobileNetV2的前8层,因为第8层的输出通道数为320,而从第9层开始,输出通道增加到1280,通道数的增加

会消耗更多的计算资源;(2) 原始的 MobileNetV2 适用于图像分类任务,但不适用于语义分割任务,因为图像通过网络后得到的特征图大小是输入的 1/32,为了增大网络的输出大小,在第 7 层和第 8 层使用空洞卷积替换普通卷积,并将第 7 层的 stride 改为 1. 改进后的 MobileNetV2 网络各层结构如表 2 所示,其中新增了 rate 作为卷积的超参数,它表示空洞卷积各元素的间隔,注意当 rate = 1 时,空洞卷积退化为普通卷积,其它参数含义与表 1 相同.

表 1　原始的 MobileNetV2 网络各层结构

层号	通道数	操作	OS	t	c	n	s
1	3	conv2d	2	–	32	1	2
2	32	bottleneck	2	1	16	1	1
3	16	bottleneck	4	6	24	2	2
4	24	bottleneck	8	6	32	3	2
5	32	bottleneck	16	6	64	4	2
6	64	bottleneck	16	6	96	3	1
7	96	bottleneck	32	6	160	3	2
8	160	bottleneck	32	6	320	1	1
9	320	conv2d 1 × 1	32	–	1280	1	1
10	1280	avgpool 7 × 7	32	–	–	1	–
11	1280	conv2d 1 × 1	–	–	k	–	–

表 2　改进后的 MobileNetV2 网络各层结构

层号	通道数	操作	OS	t	c	n	s	rate
1	3	conv2d	2	–	32	1	2	1
2	32	bottleneck	2	1	16	1	1	1
3	16	bottleneck	4	6	24	2	2	1
4	24	bottleneck	8	6	32	3	2	1
5	32	bottleneck	16	6	64	4	2	1
6	64	bottleneck	16	6	96	3	1	1
7	96	bottleneck	16	6	160	3	1	2
8	160	bottleneck	16	6	320	1	1	4

2.2　空间金字塔池化

ASPP 最早在 DeepLab 语义分割模型中提出[16],原始的结构如图 2 所示,这是一种并行的多尺度卷积模块,共有 256 个通道,其中包含 4 个空洞卷积,rate 值分别为 1、6、12、18. 在本文方法中,将图 1 中 Block8 的输出特征图作为 ASPP 的输入,由于 ASPP 消耗了大量的推断时间,所以对 ASPP 优化能对网络的整体速度有很大帮助. 本文选择了对 ASPP 中的 rate 组合和通道数进行优化,其原因在于 rate 值决定了空洞卷积各元素的间隔,对卷积的效果和语义分割的准确性有影响,而通道数的大小则决定了模型参数量的多少,直接影响模型的运算时间. 优化的结果如表 3 所示,当通道数为 256 的时候,调整 rate 以及调整分支个数对网络性能没有明显影响;将通道个数从 256 降低到 128 的时候,性能稍有损失,但模型的参数量却大大减少,这样带来的好处是消耗的计算资源和计算时间减少了很多. 所以,本文选择 rate = 1、6、12、18,通道数为 128 的 ASPP.

表 3　ASPP 不同超参数的性能对比,rate 表示空洞卷积各元素的间隔,mIoU 表示平均交并比,MAdds 表示乘积累加运算数量的个数

rate	通道数	mIoU	MAdds($\times 10^9$)	耗时(GPU)
1 + 6 + 12 + 18	256	72.26%	14.4	16ms
1 + 3 + 6 + 9	256	72.24%	14.4	16ms
1 + 4 + 8 + 16	256	72.22%	14.4	16ms
1 + 6 + 12 + 18	128	71.58%	8.2	11ms
1 + 3 + 6 + 9	128	71.60%	8.2	11ms
1 + 4 + 8 + 16	128	71.48%	8.2	11ms

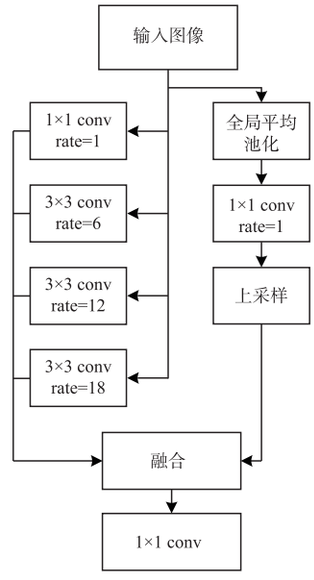


图2　原始的ASPP结构图

2.3　解码器

由于语义分割模型的输出特征图的尺寸通常远小于输入图像的尺寸,这导致网络架构无法恢复图像中细节部位的分割结果,常规的做法是使用双线性插值将预测图放大到与输入图像相同的尺寸. 本文借鉴了编码器-解码器的思想,可以将从 Block1 到 ASPP 的网络看作是一个编码器,则 ASPP 输出的特征图就是编码后的图像,解码器的作用就是对编码所得图像进行放大,并且这种放大可以恢复原图像的细节信息,本文在解码器中使用了通道注意力模块(CAB)来解决类内不连续问题. 本文设计的解码器如图 3 所示,解码器的输入分别来源于 Block7 的输出特征图和 ASPP 的输出特征图,解码器首先将 ASPP 输出的特征图上采样到与

Block7 的输出特征图相同尺寸. 同时对 Block7 做 1×1 卷积. 然后将两者连接起来,再做两次 3×3 卷积,再仿照 CAB 模块的结构,将卷积之后的结构分别与 ASPP 的输出特征图、Block7 的输出特征图做 MUL 和 SUM 操作,最终输出一个细化放大后的特征图.

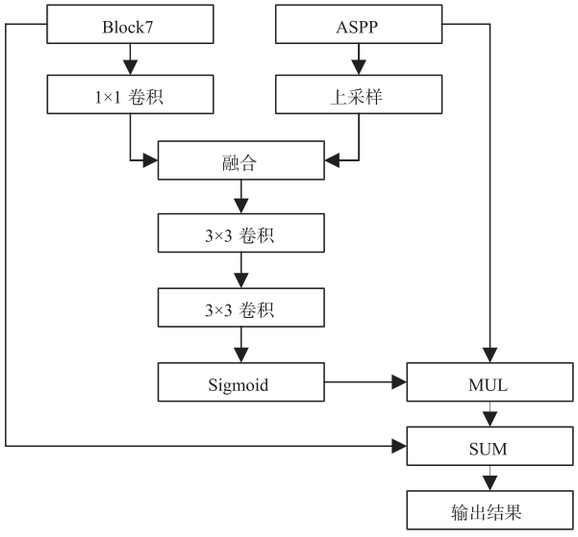


图3 解码器模块结构图

2.4 多级图像输入

随着图片分辨率的增加,语义分割网络模型的计算耗时也会随之增加,这使得模型对于高分辨率图像的语义分割速度无法满足实时性. 为解决这一问题,本文借鉴了快速语义分割网络 ICNet[28] 中的多级图像输入的思想,设计了图像多级输入的方式来减少时间消耗. 具体的多级图像输入策略如图 1 所示,首先将半尺寸的图像送入网络,使其通过 MobileNetV2、ASPP、解码器三个模块,得到 OS = 16 的特征图结果;将全尺寸的图像仅通过 Block1、Block2、Block3、Block4,这样可以得到细节比较丰富,OS = 4 的特征图结果. 将两者特征图融合就可以很大程度上恢复因为图像缩小造成的细节损失,特征图融合的过程如图 4 所示,将解码器的输出进行上采样使其 OS 从 16 减小为 4,再与 Block4 的输出进行融合,然后经过 3×3 卷积再与上采样后的特征图对应像素相加.

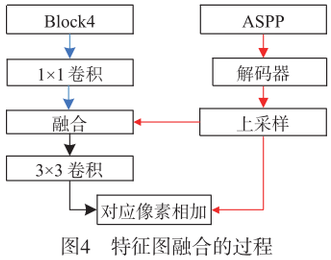


图4 特征图融合的过程

3 实验与结果

本文实验软、硬件配置如表 4 所示. 实验中,所有对比方法均在同一平台下运行.

表 4 实验软、硬件配置

配件	型号
CPU	Ryzen 7 1700X,8 核,主频 3.4GHz
内存	DDR4 32GB
GPU	NVIDIA GeForce GTX 1080
硬盘	250GB SSD
操作系统	Ubuntu16.04
开发工具	Tensorflow1.13[29],python3.6,CUDA10

本文实验共分为三个部分:(1)本文模型训练过程中的损失函数、mIoU;(2)本文模型与经典模型的定量对比;(3)本文模型与经典模型的定性对比. 实验所采用的数据集为公开数据集 VOC2012 和 Cityscapes.

3.1 定量评估指标

本文实验过程中,所使用的定量指标有三个,分别是:平均交并比(mIoU)、乘积累加运算数(MAdds)、时延(latency).

(1)平均交并比(mIoU)

几乎所有的基于深度学习的语义分割模型都使用 mIoU 作为标准的准确率度量方法. 它是分别对每个类计算(真实标签和预测结果的交并比)IOU,然后再对所有类别的 IOU 求均值. 交并比就是预测区域和实际区域的交集除以两者的并集. mIoU 的计算公式如式(1)所示.

$$\mathrm{mIoU}(I) = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k}p_{ij}+\sum_{j=0}^{k}p_{ij}-p_{ii}} \tag{1}$$

其中 I 表示输入的图像,k 是类别总数,因为通常语义分割中都会存在一个背景类,所以用 $k+1$,p_{ij} 是错误地把真实类别为 i 的像素预测为类别 j 的总个数,p_{ii} 是正确地把真实类别为 i 的像素预测为类别 i 的总个数.

(2)乘积累加运算数(MAdds)

MAdds 计算整个网络前向转播所需要的乘积累加操作的数量. CPU 和 GPU 执行一条浮点数运算的指令周期数是固定的,所以计数这种运算的数量便可以对网络前向运行的时间有一个定量的估计. 对于卷积层 MAdds 的计算式如式(2)所示.

$$\mathrm{MAdds}(L) = (C_i \times K^2 - 1) \times H \times W \times C_o \tag{2}$$

其中 L 表示卷积层,C_i 表示卷积层输入通道的个数,K 表示卷积核的大小,由于卷积核通常是正方形的,所以这里直接用 K^2 来表示卷积核的参数数量,H 和 W 分别表示 2D 卷积输出特征图的高和宽,C_0 表示卷积层输出

通道的个数.

(3)耗时(latency)

在表4 所列的实验环境下,使用模型对图像进行语义分割的平均耗时.

3.2　不同多级图像输入的对比实验

对于多级图像输入而言,选择有很多种,例如:可以仅输入全尺寸图像、可以仅输入半尺寸图像、可以仅输入四分之一尺寸图像、可以输入全尺寸和半尺寸图像、可以输入全尺寸图像和四分之一尺寸图像,等等. 本文将这些组合分为两类,一类是单尺寸输入,数据通过的路径是图1 的黑色线条和红色线条,另一类是组合尺寸输入数据通过的路径是图 1 的黑色线条、蓝色线条、红色线条. 为了选出最优的组合,本文对各种选择进行了对比实验,实验结果如表 5 所示.

表 5　不同图像输入策略对性能的影响对比

输入策略	mIoU	MAdds($\times 10^9$)	耗时(GPU)
全尺寸	72.2%	14.4	16ms
半尺寸	64.6%	3.6	4ms
四分之一尺寸	56.2%	0.9	0.96ms
全尺寸 + 半尺寸	71.3%	6.2	7ms
全尺寸 + 四分之一尺寸	64.28%	3.3	3.5ms
半尺寸 + 四分之一尺寸	63.7%	2.0	1.3ms

由表5 可知,直接使用半尺寸图像输入虽然速度提升了,但 mIoU 严重下降;使用“全尺寸 + 半尺寸”的多级输入策略可以极大地节省推断时间,同时 mIoU 只是有小幅的下降. 因此,本文选用“全尺寸 + 半尺寸”的多级图像输入策略,既保证了 mIoU,又减少了参数,提高了运算速度,参数减少的主要原因是全尺寸的图像通过的路径(黑色线条)相对较短,比较复杂的运算都使用半尺寸图像(红色线条).

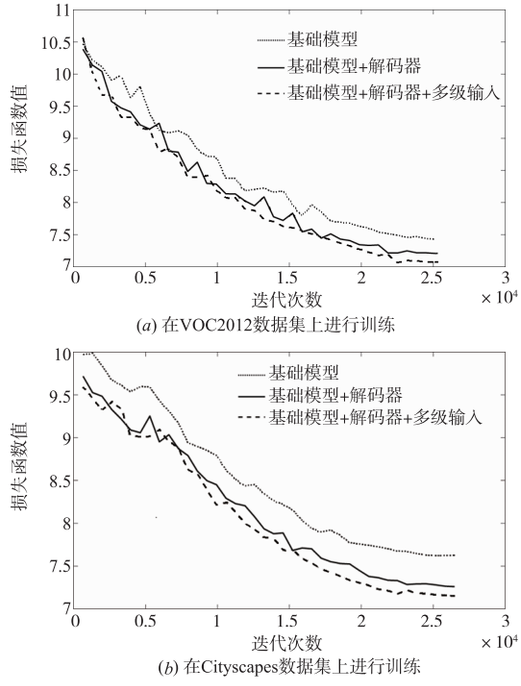
3.3　本文模型训练过程中的损失函数、mIoU

为了更好的体现本文模型各个模块的作用,实验中将模型分为三类,分别是:基础模型、基础模型 + 解码器、基础模型 + 解码器 + 多级输入. 在实验中对这三类模型分别进行了比较和分析.

本文设计的网络模型在 VOC 2012 和 Cityscapes 数据集上进行训练的时候,其损失函数随迭代次数的变化而变化,如图 5 所示. 从变化曲线中可以看出,三个模型在整个训练过程中的损失值不断下降,并在 20000 步之后趋于稳定,其中基础模型的最终损失值最大,而带有解码器和多级输入的模型最终损失值最小,这说明本文所提出的解码器和多级输入模型对模型的性能有较大提升.

本文设计的网络模型在 VOC 2012 和 Cityscapes 数据集上进行训练的时候,其 mIoU 随迭代次数的变化,如图 6 所示. 从图 6(*a*)可以看到,在 VOC2012 数据集上,基础模型达到了 72.2% 的 mIoU;在此基础上,通过解码器用很小的计算代价使 mIoU 提升到 74.4%;而引入图像多级输入仅使模型的 mIoU 下降到 73.7%,但能极大地减少推断所需的时间,节省大量的计算资源. 图 6(*b*)呈现了类似的结论,在 Cityscapes 数据集上,本文的基础模型达到了 68.5% 的 mIoU;通过解码器将 mIoU 提升到 71.6%;通过图像多级输入加快了推断的速度,而 mIoU 达到了 71.1%.

在本实验软、硬件配置下,本文设计的语义分割模型,训练需要 8 个小时.



(*a*) 在VOC2012数据集上进行训练

(*b*) 在Cityscapes数据集上进行训练

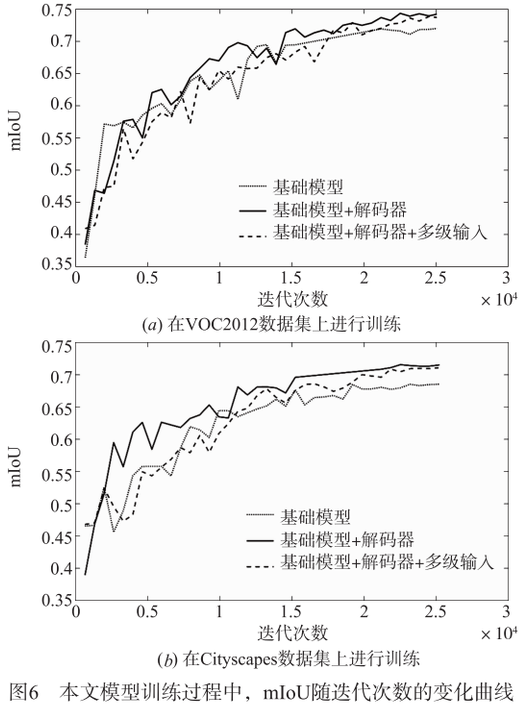
图5　本文模型训练过程中，损失函数随迭代次数的变化曲线

3.4　本文模型与经典模型的定量对比

在本实验中,将本文所设计的三个语义分割模型与近年来提出的 5 个语义分割模型(FCN[12]、DeepLabV3[17]、SegNet[21]、PSPNet[18]、DFN[21]),在速度和性能方面进行定量指标的对比. 表 6 和表 7,分别展示了在 VOC 2012 数据集和 Cityscapes 数据集下,各个语义分割模型在 mIoU、浮点运算数和推断消耗时间上的对比.

表 6 是几种网络模型在 VOC 2012 数据集上的对比,输入图像分辨率为 512 × 512,本文提出的三个语义分割模型在 mIoU 指标方面排在第 4、5、6 名,虽然在准确性方面本文模型不是最好的,但只有本文模型达到了实时处理(> 30fps)的要求. 通过对比发现本文模型的 MAdds 远小于其他的语义分割模型. PSPNet 是表中 mIoU 最高的,但 MAdds 高达 1520×10^9,推断每帧图像

所需要的平均时间更是达到了 1.7s,而本文设计的带有图像多级输入的网络能够在 512×512 分辨率的图像输入下,达到 111fps 的实时帧率,仅次于 DFANet[30],mIoU 比排名第一的 PSPNet 低 8.5%. 对比表 6 的第二行、第三行,可以看到增加了“多级输入”模块后,本文算法的 fps 从 52 增加到 111,执行速度提高了 113.46%,这是由于“多级输入”模块使得 512×512 分辨率的输入图像并没有经过整个网络结构中的所有部分,仅仅经过了 Block1 模块、Block2 模块、Block3 模块、Block4 模块以及特征融合模块(如图 1 所示),取而代之的用较低分辨率的 256×256 分辨率的降采样图像经过其他 Block5 模块、Block6 模块、Block7 模块、Block8 模块、ASPP 模块以及解码器模块,这一策略使得整个模型在运算过程中可以节约大量的时间,并且由于“特征融合”模块的作用,使得 mIoU 也只是降低了 0.3%. 表 7 是几种网络模型在 Cityscaps 数据集上的对比,输入图像分辨率为 1024×512. 随着图像分辨率的提高,可以看到各个模型所消耗的计算资源和时间都成倍增加,而本文模型所采用的图像多级输入模块,是专门针对高分辨率图像进行优化的,因此本文设计的带有解码器和多级输入的模型在 1024×512 分辨率的图像输入下,能够保证 55.6 的实时帧率,mIoU 比准确率排名第一的 PSPNet 低 4.3%.



(a) 在VOC2012数据集上进行训练

(b) 在Cityscapes数据集上进行训练

图6 本文模型训练过程中,mIoU随迭代次数的变化曲线

通过对比可以看出,本文所提出的语义分割模型,虽然准确率比 PSPNet 略有降低,但所用的计算资源仅为 PSPNet 的 0.47%,所消耗的时间仅为 PSPNet 的 0.53%;虽然本文算法的每帧耗时高于 DFANet[30]、ICNet[31]、ESNet[32]这三个最新的实时语义分割网络,但也达到了实时帧率的标准(>30fps),且本文算法的准确率也高于 DFANet[30]、ICNet[31]、ESNet[32]. 因此,在计算资源紧张或实时性要求较高的情况下,本文模型更为适宜.

表 6 本文模型与六个经典模型在 VOC 2012 数据集下之间的对比

模型名称	mIoU	MAdds ($\times10^9$)	耗时(ms)	fps
Ours(基础模型)	72.2%	14.4	16	62
Ours(基础模型+解码器)	76.4%	16.1	19	52
Ours(基础模型+解码器+多级输入)	76.1%	7.2	9	111
FCN[12]	62.2%	181	202	5
DeepLabV3[17]	77.2%	80	89	11
SegNet[21]	59.9%	31	34	29
PSPNet[18]	82.6%	1520	1692	0.6
DFN[20]	79.3%	562	624	1.6
DFANet[30]	73.4%	1.7	3	333

表 7 本文模型与八个经典模型在 Cityscapes 数据集下之间的对比

模型名称	mIoU	MAdds ($\times10^9$)	耗时(ms)	fps
Ours(基础模型)	69.5%	28.9	32	31
Ours(基础模型+解码器)	74.6%	32.2	39	25.6
Ours(基础模型+解码器+多级输入)	74.1%	14.3	18	55.6
FCN[12]	59.3%	362	404	2.5
DeepLabV3[17]	74.2%	161	179	5.6
SegNet[21]	56.7%	62	68	14.7
PSPNet[18]	78.4%	3039	3384	0.3
DFN[20]	78.3%	1124	1248	0.8
DFANet[30]	71.3%	3.4	6	166
ICNet[31]	67.7%	15.6	8.25	120
ESNet[32]	70.7%	3.3	4	249

4 结论

为解决传统语义分割算法计算资源消耗大、计算时间长、实时性差的问题,本文对传统的金字塔语义分割模型进行了改进,引入了 MobileNetV2、空间金字塔池化、解码器、多级图像输入,在略微降低语义分割准确性的情况下,上百倍的节省语义分割的计算资源消耗、上百倍的减少语义分割的计算时间. 在数据集 VOC2012、Cityscape 上进行的实验结果表明,本文模型的 mIoU 分别为 76.1%、74.1%,仅比 PSPNet 下降 8.5%、4.3%,但本文模型所消耗的计算资源仅为 PSPNet 的 0.47%,

本文模型的计算时间仅为 PSPNet 的 0.53%.这表明在计算资源紧张或实时性要求较高的情况下,本文模型更为适宜.但本文算法也存在一定的局限性,分割的准确率低于目前最佳的语义分割模型,无法在既要求实时性又要求准确性的场合应用,这也是我们下一步要解决的问题.

参考文献

[1] 陈鸿翔. 基于卷积神经网络的图像语义分割[D]. 杭州:浙江大学,2016.

[2] Thoma M. A survey of semantic segmentation[J]. IEEE Access,2016,5:1 - 11.

[3] 陈天华,郑司群,于峻川. 采用改进 DeepLab 网络的遥感图像分割[J]. 测控技术,2018,37(11):40 - 45.
CHEN Tianhua,ZHEN Siqun,YU Junchuan. Remote sensing image segmentation based on improved deeplab network[J]. Measurement and Control Technology,2018,37(11):40 - 45. (in Chinese)

[4] Siam M,Elkerdawy S,Jagersand M,et al. Deep semantic segmentation for automated driving:Taxonomy,roadmap and challenges[J]. IEEE Access,2017,1:1 - 8.

[5] 李琳辉,钱波,连静,等. 基于卷积神经网络的交通场景语义分割方法研究[J]. 通信学报,2018,39(4):123 - 130.
LI Linhui,QIAN Bo,LIAN Jing,et al. Study on traffic scene semantic segmentation method based on convolutional neural network[J]. Journal of Communications,2018,39(4):123 - 130. (in Chinese)

[6] Zeng Z,Xie W,Zhang Y,et al. RIC-Unet:Animproved neural network based on Unet for nuclei segmentation in histology images[J]. IEEE Access,2019,7:21420 - 21428.

[7] 林志伟,涂伟豪,黄嘉航,等. 深度语义分割的无人机图像植被识别[J]. 山地学报,2018,36(6):953 - 963.

[8] Malmberg F,Lindblad J,Sladoje N,Nystrom I. A graph-based framework for sub-pixel image segmentation[J]. Theoretical Computer Science,2011,412(15):1338 - 1349.

[9] Mahipal S C,Rajiv K. Performance Analysis of fuzzy C-means clustering methods for MRI image segmentation[J]. Procedia Computer Science,2016,89:749 - 758.

[10] Geman S. Stochastic relaxation,Gibbs distribution,and Bayesian restoration of images[J]. IEEE Trans Pattern Anal Mach Intell,1984,6(6):721 - 741.

[11] Caesar H,Uijlings J,Ferrari V. Region-based semantic segmentation with end-to-end training[J]. IEEE Access,2016,4:1 - 13.

[12] Long J,Shelhamer E,Darrell T. Fullyconvolutional networks for semantic segmentation[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence,2014,39(4):640 - 651.

[13] Simonyan K,Zisserman A. VeryDeep Convolutional Networks for Large-scale Image Recognition[EB/OL]. https://arxiv.org/abs/1409.1556,2014.

[14] Szegedy C,Liu W,Jia Y,et al. Going Deeper With Convolutions[EB/OL]. https://arxiv.org/abs/1409.4842,2014.

[15] Yu F,Koltun V. Multi-scale Context Aggregation by Dilated Convolutions[EB/OL]. https://arxiv.org/abs/1511.07122,2014.

[16] Ronneberger O,Fischer P,Brox T. U-Net:convolutional networks for biomedical image segmentation[J]. IEEE Access,2015,11:1 - 10.

[17] He K,Zhang X,Ren S,et al. Deepresidual learning for image recognition[J]. IEEE Access,2015,9:1 - 16.

[18] Chen L C,Papandreou G,Kokkinos I,et al. DeepLab:semantic image segmentation with deep convolutional nets,atrous convolution,and fully connected CRFs[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence,2016,40(4):834 - 848.

[19] Chen L C,Papandreou G,Schroff F,et al. Rethinking Atrous Convolution for Semantic Image Segmentation[EB/OL]. https://arxiv.org/abs/1706.05587,2017.

[20] Zhao H,Shi J,Qi X,et al. Pyramidscene parsing network[J]. IEEE Access,2016,9:1 - 13.

[21] Hu J,Shen L,Albanie S,et al. Squeeze-and-excitation Networks[EB/OL]. https://arxiv.org/abs/1709.01507,2019.

[22] Yu C,Wang J,Peng C,et al. Learning adiscriminative feature network for semantic segmentation[J]. IEEE Access,2018,3:1 - 11.

[23] Badrinarayanan V,Kendall A,Cipolla R. SegNet:a deep convolutional encoder-decoder architecture for image segmentation[J]. IEEE Access,2015,7:1 - 18.

[24] He K,Zhang X,Ren S,et al. Identitymappings in deep residual networks[J]. IEEE Access,2016,11:1 - 9.

[25] Huang G,Sun Y,Liu Z,et al. Deepnetworks with stochastic depth[J]. IEEE Access,2016,4:1 - 8.

[26] Howard A G,Zhu M,Chen B,et al. MobileNets:efficient convolutional neural networks for mobile vision applications[J]. IEEE Access,2017,6:1 - 14.

[27] Sandler M,Howard A,Zhu M,et al. Mobile NetV2:inverted residuals and linear bottlenecks[J]. IEEE Access,2018,5:1 - 12.

[28] Zhao H,Qi X,Shen X,et al. ICNet forreal-time semantic segmentation on high-resolution images[J]. IEEE Access,2017,3:1 - 21.

[29] Abadi,Martín,Agarwal,Ashish,Barham,Paul,Brevdo,et al. TensorFlow:large-scale machine learning on heterogeneous distributed systems[J]. IEEE Access,2016,7:1 - 46.

[30] Hanchao Li,Pengfei Xiong,Haoqing Fan,Jian Sun. DFA-

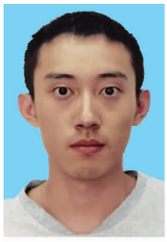
Net: Deep Feature Aggregation for Real-time Semantic Segmentation [EB/OL]. https://arxiv.org/abs/1904.02216,2019.

[31] Hengshuang Zhao, XiaoJuan Qi, Xiaoyong Shen, Jianping Shi, Jiaya Jia. ICNet forreal-time semantic segmentation on high-resolution images[A]. 15th European Conference on Computer Vision[C]. New York: IEEE, 2018. 418-434.

[32] Yu Wang, Quan Zhou, Xiaofu Wu. ESNet: An Efficient Symmetric Network for Real-time Semantic Segmentation [EB/OL]. https://arxiv.org/abs/1906.09826,2019.

作者简介

孟 琭 男,1982 年 2 月出生于辽宁沈阳. 现为东北大学信息科学与工程学院副教授. 主要研究方向为人工智能、图像处理.
E-mail: menglu@ise.neu.edu.cn



徐 磊 男,1997 年 11 月出生于辽宁大连. 研究生. 主要研究方向为计算机视觉.
E-mail: 19s001019@stu.hit.edu.cn

郭嘉阳 男,1985 年生于福建厦门. 现为美国辛辛那提大学电气工程与计算机系博士后,IEEE 会员. 主要研究方向为人工智能,机器学习,医学图像处理,信号处理等.
E-mail: guojy@mail.uc.edu