第 58 卷 第 16 期/2021 年 8 月/激光与光电子学进展 研究论文

激光与光电子学进展 先进成像

基于卷积神经网络的轻量化目标检测网络

程叶群[1,2]，王艳[1,2]，范裕莹[1,2]，李宝清[1*]

[1]中国科学院上海微系统与信息技术研究所微系统技术重点实验室，上海 201800；

[2]中国科学院大学，北京 100049

摘要 针对目前常用的目标检测算法计算复杂度高，在嵌入式平台检测速度低的问题，提出一种适用于嵌入式平台的轻量化目标检测网络(BENet)。首先，该网络在 MobileNetv2 轻量化网络的基础上加入通道特征交织模块，来设计骨干网络，有效地增强了轻量化骨干网络的特征表达；其次，提出自适应多尺度加权特征融合模块，通过对不同尺度的特征进行权重分配，学习各个尺度特征之间的相关性；最后，尝试引入空间金字塔池化结构来获取不同感受野的上下文信息。在 VOC 数据集上的实验结果表明：所提 BENet 在保持较高目标检测精度和检测速度的同时，具有较低的计算复杂度和较小的参数量，更适合应用于嵌入式平台。

关键词 图像处理；目标检测；轻量化网络；通道特征交织；特征融合

中图分类号 TP391.4 **文献标志码** A **doi**：10.3788/LOP202158.1610023

Lightweight Object Detection Network Based on Convolutional Neural Network

Cheng Yequn[1,2], Wang Yan[1,2], Fan Yuying[1,2], Li Baoqing[1*]

[1]*Key Laboratory of Microsystem Technology, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai* 201800, *China*;

[2]*University of Chinese Academy of Sciences, Beijing* 100049, *China*

Abstract Considering the high computational complexity and low detection speed of the common object detection algorithms on an embedded platform, this study proposes a lightweight object detection network (BENet) suitable for embedded platforms. First, the proposed network added a channel feature interweaving module to the MobileNetv2 lightweight network to design the backbone network, which effectively enhanced the feature expression of the lightweight backbone network. Second, an adaptive multiscale weighted feature fusion module was proposed to learn the correlation between the features with various scales by assigning weights to the features with different scales. Finally, we attempted to introduce a spatial pyramid pooling structure to obtain the context information of different receptive fields. The experimental results on the VOC dataset show that the proposed BENet maintains high object detection accuracy and speed while has lower computational complexity and smaller parameters. Additionally, it is more suitable for embedded platforms.

Key words image processing; object detection; lightweight network; channel feature interweaving; feature fusion

OCIS codes 100.3008; 040.1880; 150.0155

1 引 言

目标检测技术作为计算机视觉领域的基本技术之一，已广泛应用于智能交通、视频监控、自动驾驶等领域[1]。近年来，随着深度学习技术的不断发展，卷积神经网络凭借强大的特征表达能力和泛化能力

收稿日期：2020-09-17；**修回日期**：2020-10-13；**录用日期**：2020-10-22

基金项目：微系统技术重点实验室基金(6142804190304)

通信作者：*sinoiot@mail.sim.ac.cn

在计算机视觉领域取得了巨大的成功[2]，目标检测技术也从基于手工特征的传统算法转向了基于深度神经网络的检测算法。

目前该领域主要有两种检测算法，其中一种是双阶段（two-stage）目标检测算法，如 Faster-RCNN[3]、Mask-RCNN[4]等。这类算法首先生成一系列样本的候选框，然后通过卷积神经网络进行分类和回归，最终可以达到很高的准确率，但是这类算法的计算复杂度较高。另一种是单阶段（one-stage）目标检测算法，如 SSD[5]和 YOLO[6]系列算法。此类算法不需要生成区域候选框，直接通过卷积神经网络来预测物体的类别和位置，因此该类算法通常具有更快的检测速度[7]。由于嵌入式平台的计算能力和内存资源有限，不适合部署大型网络，因此这些算法模型难以在嵌入式平台上得到广泛应用，妨碍了目标检测技术在工业上的应用普及。目前在嵌入式平台上普遍使用的是 YOLO 和 SSD 算法的小型化版本，例如文献[8]通过更换基础网络，提出了一种基于 SSD 的快速检测方法；文献[9]通过缩减 YOLOv3 骨干网络的残差单元数和通道数来提高网络的检测速度；文献[10]通过将标准卷积替换为深度可分离卷积来减少网络参数量。虽然这些算法计算量低，检测速度快，但是检测精度普遍偏低，难以实现计算量和精度的平衡。

为了在降低计算量和参数量的同时，保持较高的检测精度和速度，本文借鉴 YOLOv3 算法，提出一种新的适用于嵌入式平台的轻量化目标检测网络（BENet）。在特征提取阶段，基于轻量化网络 MobileNetv2 的设计，利用深度可分离卷积大幅度减少了参数量和计算量[11]，然后对通道特征交织思想[12]和深度可分离卷积思想进行结合，设计出增强的轻量化特征提取网络（I-MobileNetv2），并在骨干网络最后加入空间金字塔池化（SPP）结构[13]，以获得不同感受野的语义信息。在检测阶段，为了提高网络的多尺度目标检测的性能，提出了自适应多尺度加权特征融合模块，该模块可以学习不同尺度特征之间的相关性，使得深层的语义特征和浅层的细粒度特征更好融合。实验结果显示：所提网络模型在保持较高目标检测精度和检测速度的同时，计算复杂度较低，从而实现了目标检测精度和计算量的良好平衡。

2 轻量级神经网络

为了在满足嵌入式设备存储空间和功耗要求的同时，提高目标检测的效率和能力，近年来轻量级神经网络架构的设计受到了广泛的关注。对于自动驾驶或辅助驾驶系统等对实时处理能力要求较高的应用场景，平衡好网络的检测速度和精度是至关重要的。单阶段目标检测算法因效率高、结构简单被广泛应用。目前常见的轻量化目标检测模型通常采用单阶段目标检测算法和高效的轻量级骨干网络，将分类问题中的训练模型作为目标检测的预训练模型，这可以为训练检测器提供更丰富的语义信息。

人工设计轻量级神经网络的主要思想是通过优化卷积的计算方法设计出更高效的网络结构。例如采用深度可分离卷积、分组卷积等轻量卷积方式，可以有效减少卷积计算过程中的计算量。对于轻量化的网络设计，目前常用的有 SqueezeNet[14]、MobileNet[14]和 ShuffleNet[15]等结构。SqueezeNet 采用精心设计的压缩再拓展模块，有效降低了卷积计算量；MobileNet 系列充分发挥了深度可分离卷积的优势，通过引入逆残差结构，提升了卷积计算的效率；ShuffleNet 系列在分组卷积的基础上引入了通道混洗操作，避免了大量的 1×1 卷积操作。人工设计轻量级神经网络通过引入更加高效的卷积单元，在不损失网络性能的条件下，降低了网络参数量，提升了网络的计算速度。

除了人工设计轻量级神经网络，近年来神经网络架构搜索（NAS）技术也取得了一定的成果。NAS 实现了轻量化神经网络的自动化构建。另外模型剪枝、权值共享、知识蒸馏等方法也被用来进一步实现神经网络模型的压缩。

3 BENet 算法原理

3.1 增强的轻量化骨干网络

在目前常用的目标检测算法中，骨干网络大多采用 ResNet101、DarkNet53 这类深度残差网络。这类深度残差网络由于使用多层卷积残差单元，虽然可以提取有效的特征信息，但存在网络参数过多和网络计算量大的问题。为了降低网络的参数量和计算量，从而提高网络的检测速度，所提目标检测模型的骨干网络基于 MobileNetv2 的设计。MobileNetv2 是一种轻量级的骨干网络，通过将常规卷积分解为通道卷积（depthwise convolution）和点卷积（pointwise convolution）[14]，可以有效降低网络的计算复杂度和模型参数量。通道卷积对输入向量的每一个通道进行卷积运

算，一个卷积核负责一个通道，之后使用点卷积在深度方向上对上一步的特征图进行加权组合，生成新的特征图，最终得到和传统卷积相同的结果，如图 1 所示。

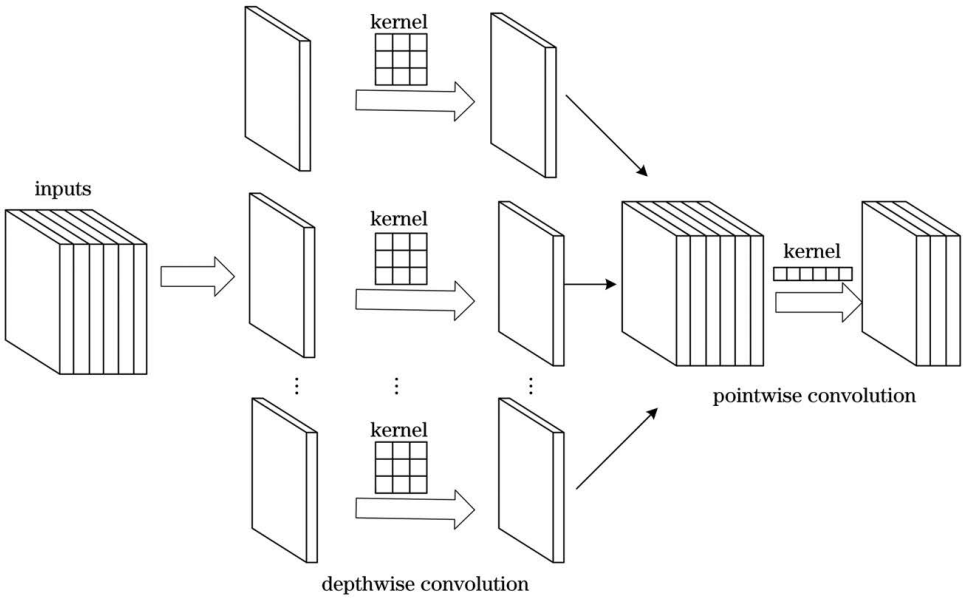

图 1　深度可分离卷积

Fig. 1　Depthwise separable convolution

假设输入特征图的尺寸为 $D_F \times D_F \times M$，标准卷积核的尺寸为 $D_K \times D_K \times M$，输出特征图的尺寸为 $D_F \times D_F \times N$，输入和输出通道数分别为 M 和 N。当存在 padding 且步长为 1 时，标准卷积层的计算量为

$$N_{SC} = D_K \times D_K \times M \times N \times D_F \times D_F。 \quad (1)$$

通道卷积负责滤波，卷积核尺寸为 $D_K \times D_K \times 1$，共 M 个；点卷积负责转换通道，卷积核尺度为 $1 \times 1 \times M$，共 N 个。因此深度可分离卷积的计算量为

$$N_{DSC} = D_K \times D_K \times M \times D_F \times D_F + M \times N \times D_F \times D_F。 \quad (2)$$

对两者的计算量进行比较，由于在实际应用中通常采用 3×3 卷积，而且输出通道数 N 一般较大，因此深度可分离卷积的计算量可缩小为标准卷积的 1/9 左右。

$$\frac{N_{DSC}}{N_{SC}} = \frac{D_K \times D_K \times M \times D_F \times D_F + M \times N \times D_F \times D_F}{D_K \times D_K \times M \times N \times D_F \times D_F} = \frac{1}{N} + \frac{1}{D_K^2}。 \quad (3)$$

MobileNetv2 作为分类网络，只能在单一的特征图上作预测，因此网络最后生成的特征图通常具有低分辨率、大感受野的特性。但是在目标检测中，需要在不同表达能力的特征图上进行预测，因此高分辨率的空间信息对准确的目标定位同样具有重要作用。为了解决逐层前向传递可能出现信息丢失和浅层特征表达能力不足的问题，将通道特征交织思想应用于骨干网络的设计中，在 MobileNetv2 骨干网络的基础上加入通道特征交织模块。对同一层内不同通道组之间的特征图进行连接，不仅重用了该层的部分信息，还使得同一层的特征图包含了不同感受野的特征，增强了轻量化骨干网络的多尺度特征表达能力。通道特征交织模块按照通道数将第一层 1×1 卷积层的输出特征图均分为 s 组特征，均分后的每一组特征用 x_i 表示，$i \in \{1,2,3,\cdots,s\}$。对于分组后的每一组特征，除了第一组特征，其他组特征都会对上一组的输出特征 y_{i-1} 与当前组的特征 x_i 进行残差连接，通过卷积操作后生成新的特征表达。每一组的输出 y_i 为

$$y_i = \begin{cases} x_i, & i = 1 \\ \mathrm{Conv}(x_i + y_{i-1}), & 1 < i \leqslant s \end{cases}, \quad (4)$$

式中：Conv 为卷积操作，卷积核大小为 3×3；s 为通道的分组数，设置为 4。通道特征交织模块结构如图 2 所示。

在骨干网络的最后 4 次下采样之前加入通道特

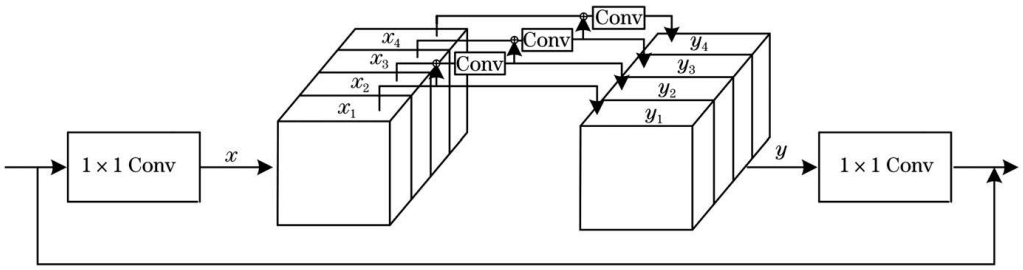

图 2　通道特征交织模块

Fig. 2　Channel feature interweaving module

征交织模块，通过对同一卷积层的特征进行通道特征交织，不仅重用了上面一层的部分信息，还使得每一组卷积的输出特征包含了丰富的多尺度特征信息，有效地解决了轻量化骨干网络特征提取能力不足的问题。改进后的骨干网络 I-MobilenNetv2 的相关参数如表 1 所示。

表 1　I-MobileNetv2 的相关参数

Table 1　Related parameters of I-MobileNetv2

Input size	Operator	Stride	N	Output size
416×416×3	Conv2d	2	1	208×208×32
208×208×32	Bottleneck	1	1	208×208×16
208×208×16	Bottleneck	2	2	104×104×24
104×104×24	Feature interweaving	1	2	104×104×24
104×104×24	Bottleneck	2	3	52×52×32
52×52×32	Feature interweaving	1	2	52×52×32
52×52×32	Bottleneck	2	4	26×26×64
26×26×64	Bottleneck	1	3	26×26×96
26×26×96	Feature interweaving	1	2	26×26×96
26×26×96	Bottleneck	2	3	13×13×160
13×13×160	Bottleneck	1	1	13×13×320
13×13×320	Feature interweaving	1	2	13×13×320
13×13×320	Conv2d	1	1	13×13×1280

3.2　自适应多尺度加权特征融合模块

为了检测不同尺寸的目标，基于卷积神经网络的目标检测算法大多采用多尺度输出和特征融合的策略，例如 SSD、YOLOv3 等。其中特征网络金字塔(FPN)结构是目前最常用的一种特征融合方式。FPN 采用一种自上而下、横向连接的方式融合两个相邻尺度的特征，通过对具有低分辨率、强语义信息的特征进行上采样，再与具有高分辨率、细粒度的特征相结合，可以有效提高目标检测的精度[16]。然而 FPN 也存在一些不足。首先，FPN 这种自上而下的融合方式仅仅考虑了两个相邻尺度的特征，没有利用其他尺度的特征；其次，之前一些常见的特征融合方式没有对不同尺度特征的重要性进行区分，不同分辨率的输入特征的作用不同，而且对融合后输出特征的贡献也是不同的。

针对上述问题，为了更好地进行特征融合操作，使用自适应多尺度加权特征融合模块来改进 FPN，通过在融合过程中为每个尺度输入特征增加一个额外的权重，来学习每个尺度输入特征的重要性。

图 3 为自适应多尺度加权特征融合的过程，由于对三个不同尺度进行检测，各分支具有不同的分辨率和通道数，因此对每一尺度进行特征融合之前需要对各个特征图的大小和通道数进行统一。定义第 l 个尺度的特征图用 X^l 表示，$l\in\{1,2,3\}$。对于尺度 l，需要对其他尺度的特征 $X^n(n\neq l)$ 的尺寸和分辨率进行调整。对于上采样，首先利用 1×1 卷积

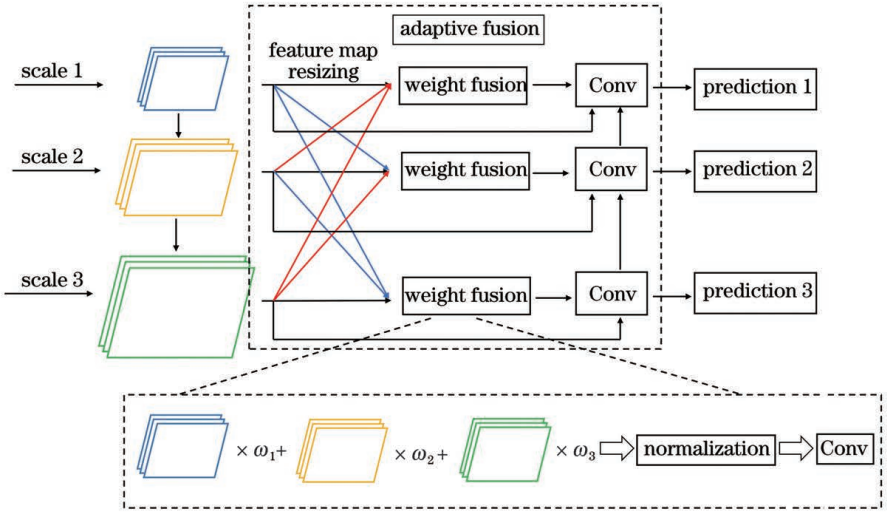

图 3　自适应多尺度加权特征融合模块

Fig. 3　Adaptive multi-scale weighted feature fusion module

将通道数变成一致，然后使用最近邻差值提高特征图分辨率；对于 2 倍的降采样，直接利用卷积操作来改变通道数并降低特征图分辨率，卷积核大小为 3×3，步长为 2；对于 4 倍降采样，需要额外增加一个步长为 2 的最大池化操作。

将不同尺度的特征图调整为统一尺寸之后，为了让网络自动学习不同尺度特征之间的重要性，需要对三个尺度的特征进行加权特征融合操作。对于尺度 l，融合后的特征 F^l 为

$$F^l=\mathrm{Conv}\left(\frac{\omega_1\cdot X^{1\to l}+\omega_2\cdot X^{2\to l}+\omega_3\cdot X^{3\to l}}{\omega_1+\omega_2+\omega_3+\xi}\right),\tag{5}$$

式中：ω_1，ω_2 和 ω_3 均为可训练参数，代表三个尺度特征的加权系数；$X^{n\to l}$ 为将尺度 n 的特征调整为尺度 l 的特征；ξ 为一个很小的浮点数，防止分母为 0。分母为加权系数之和，是为了进行归一化操作。

受 PANet[17] 的启发，为了使浅层的细粒度特征和深层的语义特征进一步融合，对加权融合后的特征增加一条自下而上的路径，并加入残差连接，重用之前的部分特征。融合后的特征输出到检测头，对目标进行分类和坐标回归。

3.3　空间金字塔池化

为了获取不同大小感受野的上下文语义信息，进一步提高模型的检测精度，在改进的骨干网络最后加入空间金字塔池化结构。传统的金字塔池化用来解决全连接层的输入必须是固定特征向量的问题，空间金字塔池化技术可以使构建的网络支持任意尺寸的图片，不需要经过裁剪和缩放操作。不同于传统的金字塔池化技术，空间金字塔池化结构主要对骨干网络输出的特征信息进行多尺度特征融合。

空间金字塔结构如图 4 所示。通过骨干网络的特征提取，骨干网络最后输出的特征图已经包含丰富的语义信息，并且特征图的大小已经从输入的 416×416×3 变成 13×13×1280。为了降低池化后数据拼接的维度，先利用 1×1 卷积将输出特征图维度降为 13×13×520，然后经过三种不同大小的采样核池化，最后对原始特征和池化后的特征在通道级进行合并。

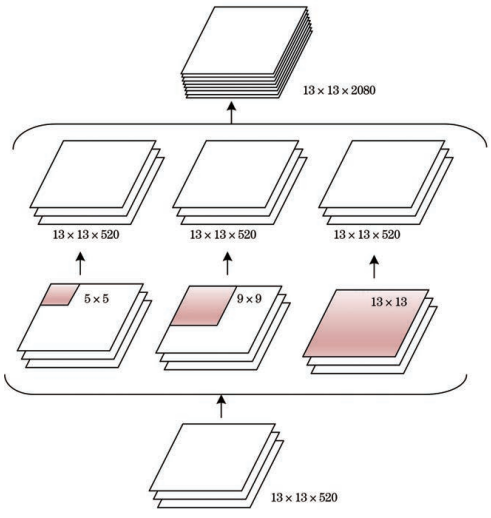

图 4　空间金字塔池化

Fig. 4　Spatial pyramid pooling

最大池化操作（Max pooling）的采样核大小对感受野的影响非常大。将池化的采样核设置为 5，9，13，最大的池化核设置为输出特征图的大小，可以获得全局特征。为了保持输出特征图的大小不变，令池化操作的步长为 1 并对特征图进行填充

(Padding)。通过 SPP 模块，池化操作实现了局部特征和全局特征的特征级融合，进一步提升了特征图的表达能力。

3.4　损失函数

所提损失函数基本遵循 YOLOv3 算法的设计，将其分为位置误差、置信度误差和分类误差。为了加快网络的收敛速度并获得更好的预测效果，引入了 GIOU loss[18] 和 Focal loss[19] 来设计 BENet 的损失函数。Intersection Over Union(IOU)在目标检测中可以用来评价预测框和真实框的距离。然而在两个框没有相交的情况下，IOU 为 0，会导致损失值为 0，无法进行梯度回传。GIOU 与 IOU 类似，也是一种距离度量，它不仅关注重叠区域，还关注其他非重叠区域，能更好地反映预测框和真实框的距离。因此利用 GIOU loss 来对预测框的位置进行回归，GIOU 可以表示为

$$d_{\mathrm{GIOU}}=\frac{|A\cap B|}{|A\cup B|}-\frac{|C\backslash(A\cup B)|}{|C|},\tag{6}$$

式中：A 为预测框；B 为真实框；C 为包含 A 和 B 的最小框。因此 GIOU loss 可以表示为

$$L_{\mathrm{GIOU}}=s_{\mathrm{bbox}}\times(1-d_{\mathrm{GIOU}}),\tag{7}$$

式中：$s_{\mathrm{bbox}}=2-w\times h/s_i^2$，$s_i$ 为 input_size，w 为目标真实框的宽，h 为目标真实框的高。s_{bbox} 可以根据检测目标的大小调整其在损失函数中的权重，可以增加小目标对象对损失函数的影响。

前景背景置信度损失 L_{conf} 包含 Focal loss 函数和二元交叉熵损失函数。Focalloss可以降低训练过程中大量简单负样本所占的权重，解决正负样本比例失衡的问题。

$$L_{\mathrm{conf}}=-\alpha_1\times(y-\hat{y})^{\gamma}\times y\log\hat{y}-\alpha_2\times(y-\hat{y})^{\gamma}\times(1-y)\log(1-\hat{y}),\tag{8}$$

式中：y 表示目标的真实置信度；$\hat{y}$ 表示预测的置信度；γ 和 α_1、α_2 均表示设定好的固定值。α_1、α_2 为 1，γ 为 2。

对于分类损失，由于多类别的预测问题可以看成多个二分类问题，分类损失函数也使用二元交叉熵损失函数。

$$L_{\mathrm{class}}=-p_{\mathrm{c}}\log\hat{p}_{\mathrm{c}}-(1-p_{\mathrm{c}})\log(1-\hat{p}_{\mathrm{c}}),\tag{9}$$

式中：p_{c} 代表目标的真实类别；$\hat{p}_{\mathrm{c}}$ 代表预测目标的类别。

因此总的损失函数为

$$L_{\mathrm{total}}=\sum_{i=1}^{3}L_{\mathrm{GIOU}}^{i}+L_{\mathrm{conf}}^{i}+L_{\mathrm{class}}^{i}。\tag{10}$$

3.5　整体的网络结构

通过引入通道特征交织模块改进的轻量化骨干网络、自适应多尺度加权特征融合模块和空间金字塔结构，建立了 BENet，如图 5 所示。通道特征交织模块和 SPP 模块有效增强了网络的感受野，改进了轻量化网络特征表达能力的不足。BENet 采用特征金字塔结构在三个尺度上分别对大、中、小的物体进行检测。为了更好地融合三个不同尺度之间的特征，提出自适应多尺度加权特征融合模块，通过在融合过程中为每个尺度输入特征增加一个额外的权重，来学习多尺度输入特征的相关性，并在三个不

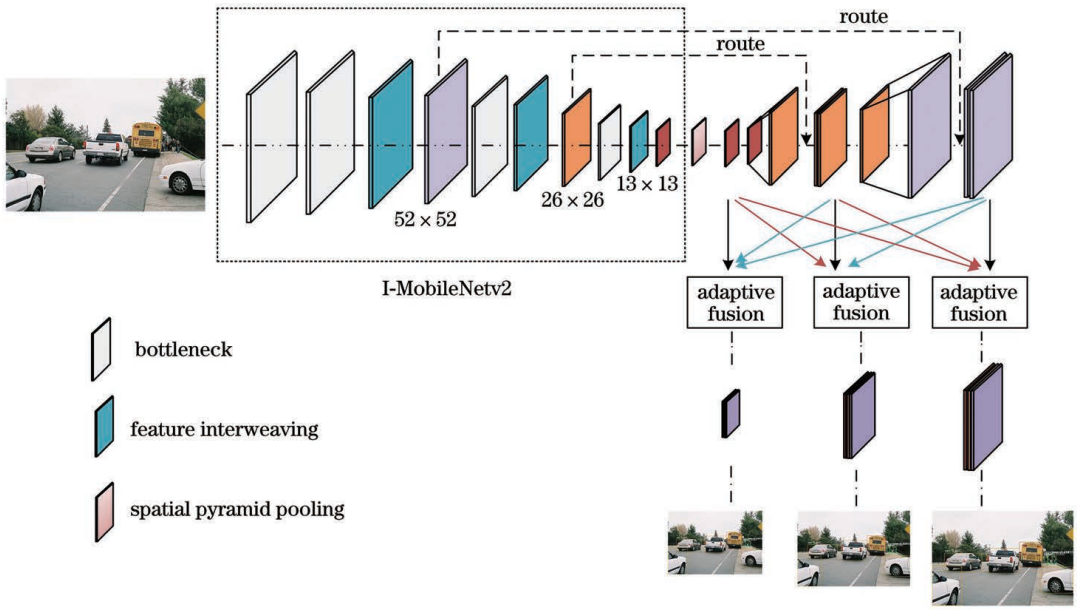

图 5　BENet 的整体网络结构

Fig. 5　Overall network structure of BENet

同尺度上进行检测输出。

4 实验结果分析

实验仿真使用 TensorFlow 框架，操作系统为 Ubuntu 14.04，CPU 为 Intel XeonE5-2620 V4，GPU 为 Nvidia GeForce GTX 1080Ti。

4.1 实验数据集

为了验证所提目标检测算法的有效性，采用目标检测领域常用的标准化数据集 PASCAL VOC。PASCAL VOC 数据集一共包含了 20 类目标，在训练阶段，将 PASCAL VOC 2007 的训练验证集和 PASCAL VOC 2012 的训练验证集作为网络的训练数据集（一共包含 16551 张图片）；在测试阶段，将 PASCAL VOC 2007 的测试集作为网络的测试数据集（一共包含 4952 张图片）。

4.2 实验细节

所有的模型均采用批量随机梯度下降法（SGD）来优化损失函数。所有模型均训练 100 个 epochs。为了使模型训练更加稳定，训练最开始的 2 个 epochs 采用“warm up”策略，初始学习率设为 0.0001，并采用 cosine learning rate 策略[20]来调整网络的学习率。数据增强采用图像旋转、随机裁剪、色彩抖动、mix-up[20]等方法。采用多尺度训练策略，每个 epoch 下，随机从预先定义的几个固定尺度中随机选择一个尺度进行训练，预先定义的尺寸为{320，352，384，416，448，480，512，544，576，608}，均为 32 的倍数。

4.3 实验结果和性能分析

采用平均精度（AP）对每一类目标的检测精度进行评估，采用均值平均精度（mAP）衡量多类目标的平均检测精度，采用每秒帧数（FPS）衡量目标检测器的速度。表 2 为所提 BENet 算法和其他目前常用的目标检测算法在 PASCAL VOC 数据集上的对比结果，各种算法的检测结果均来自相关论文，包括 Faster-RCNN[3]、SSD[5]、YOLO[6]、SSDLite[14]、DSSD[21]、R-FCN[22]和 RFBNet[23]。为了与 YOLOv3 算法[24]进行直接比较，YOLOv3、YOLOv3 tiny 和所提 BENet 采用 4.2 实验细节中相同的训练方法。训练的损失曲线如图 6 所示，横坐标代表迭代次数，纵坐标代表损失值，可以看出，BENet 的损失函数下降得更快，经过 400000 次迭代后，两者的损失值均趋于稳定。

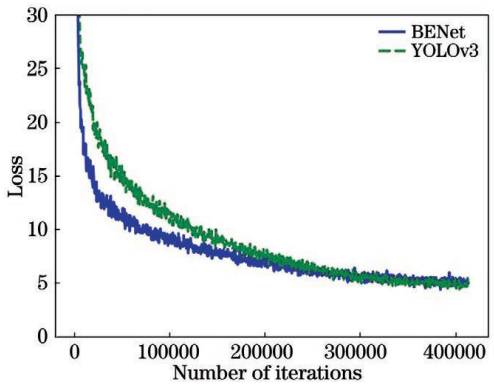

图 6 BENet 和 YOLOv3 的损失曲线

Fig. 6 Loss curve of BENet and YOLOv3

表 2 不同目标检测算法在 PASCAL VOC 数据集上的结果对比

Table 2 Comparison of the results of different object detection algorithms on PASCAL VOC dataset

Algorithm	Input size	Backbone	mAP/%	FPS	Params/10^6
Faster-RCNN	600×1000	VGG	73.2	7	138.5
SSD	512×512	VGG	76.8	22	33.1
DSSD	513×513	ResNet-101	81.5	5.5	
SSDLite	300×300	MobileNet	72.7	56	6.8
R-FCN	600×1000	ResNet-101	80.5	9	
RFBNet	512×512	VGG	82.2	38	34.5
YOLO	448×448		66.4	20	86.7
YOLOv3 tiny	544×544		61.1	116	8.5
YOLOv3	544×544	Darknet 53	80.4	26.5	62.3
BENet	544×544	I-MobileNetv2	78.4	49	7.9

根据表 2 可以发现：在检测精度方面，所提 BENet 达到 78.4%，比基于区域的目标检测算法 Faster-RCNN 高 5.2 个百分点，比基于轻量级网络的 YOLOv3 tiny 和 SSDLite，分别提高了 17.3 和

5.7 个百分点，仅比 YOLOv3 低 2 个百分点；在检测速度方面，与 YOLOv3 相比，所提 BENet 的检测速度提高近一倍，仅低于 YOLOv3 tiny 和 SSDLite；在参数量方面，所提 BENet 低于大部分目标检测网络，仅比 SSDLite 高 1.1×10^{6}。所提 BENet 在检测精度、检测速度和参数量上实现了很好的平衡，更适合应用于嵌入式平台。

YOLOv3 算法作为单阶段目标检测算法的代表，在检测精度和速度方面取得了良好的平衡。表 3 为所提 BeNet 和 YOLOv3 系列算法在模型大小和计算量方面的对比，在模型参数量上，BENet 的模型大小仅为 32 MB，约为 YOLOv3 模型大小的 $\frac{1}{8}$；在计算量上，BENet 的浮点计算数（BFLOPs）仅为 YOLOv3 算法的 10%，使得 BENet 可以在 YOLOv3 网络无法部署的嵌入式平台上运行。

表 3　BENet 和 YOLOv3 模型大小和计算量的对比

Table 3　Comparison of model size and number of calculations between BENet and YOLOv3

Algorithm	Model size /MB	BFLOPs
YOLOv3 tiny (416)	34	5.56
YOLOv3(416)	236	65.86
BENet(416)	32	6.31

为了验证 BENet 中的各个模块对实验结果的影响，在 PASCOL VOC 数据集上进行了消融实验。将 BENet 算法的骨干网络替换为 MobileNetv2 作为消融实验的 baseline，实验结果如表 4 所示。

表 4　在 PASCAL VOC 数据集上的消融实验

Table 4　Ablation experiment on PASCAL VOC dataset

BENet baseline	With MobileNetv2	With I-MobileNetv2	With SPP	With weighted feature fusion	mAP /%
√	√				75.1
√		√			76.3
√		√	√		77.6
√		√	√	√	78.4

从表 4 可以发现，改进的各个模块对网络的检测精度都有一定的提升。改进的骨干网络 I-MobileNetv2 通过加入通道特征交织模块，让输出特征图包含了丰富的多尺度信息，有效增强了网络的特征表达能力，与基于 MobileNetv2 的模型相比，提升了 1.2 个百分点；空间金字塔模块实现了局部特征和全局特征的融合，可以获得不同大小感受野的上下文信息，将 mAP 从 76.3% 提升到 77.6%；自适应加权特征融合模块可以学习不同尺度特征之间的相关性，有效利用了不同尺度之间的特征信息，将 mAP 从 77.6% 提升到 78.4%。

4.4　测试集上的实验效果图

图 7 是 BENet 和 YOLOv3、YOLOv3 tiny 在 VOC 数据集上的可视化检测结果对比。其中，图 7(a)是 YOLOv3 的检测结果，图 7(b)是 BENet 的检测结果，图 7(c)是 YOLOv3 tiny 的检测结果。通过对比可以发现，BENet 和 YOLOv3 具有相近的检测效果。

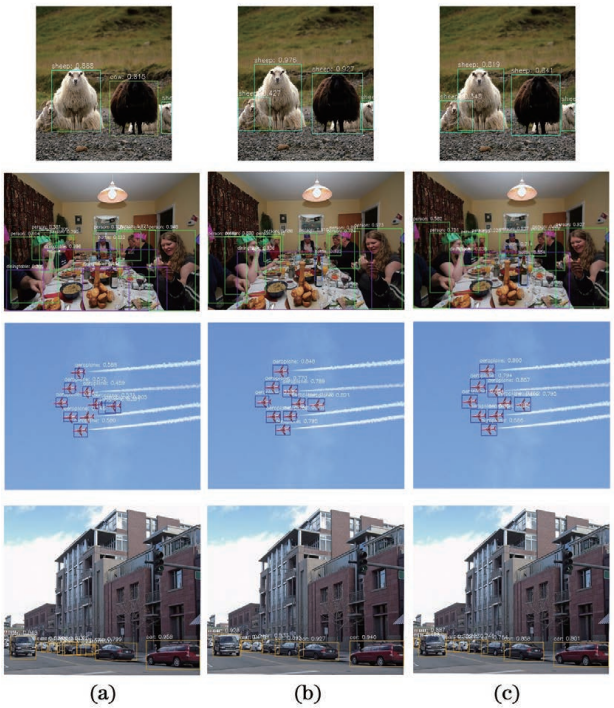

图 7　不同算法在 VOC 数据集上的检测结果对比。(a)YOLOv3；(b)BENet；(c)YOLOv3 tiny

Fig. 7　Comparison of the detection results of different algorithms on the VOC dataset. (a) YOLOv3; (b) BENet; (c) YOLOv3 tiny

5　结　　论

提出了一种适用于嵌入式平台的轻量化目标检测网络 BENet。该模型首先将通道特征交织模块

和 MobileNetv2 网络相结合，构建了新的轻量化骨干网络；然后引入空间金字塔模块，获取不同感受野的上下文信息；最后提出了自适应多尺度加权特征融合模块，使网络更好地利用多尺度的特征信息。实验结果表明：所提目标检测模型和常用的目标检测算法相比，在检测精度、检测速度和计算量上表现更加均衡，可以满足大部分嵌入式平台的需求。下一步工作将是优化模型结构，比如进行模型剪枝操作，以达到减少冗余通道、加快网络推进速度的目的。

参 考 文 献

[1] Zou Z X, Shi Z W, Guo Y H, et al. Object detection in 20 years: a survey[EB/OL]. (2019-05-16)[2020-07-28]. https://arxiv.org/abs/1905.05055.

[2] Krizhevsky A, Sutskever I, Hinton G E, et al. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84-90.

[3] Ren S Q, He K M, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[4] He K M, Gkioxari G, Dollár P, et al. Mask R-CNN[C]// 2017 IEEE International Conference on Computer Vision (ICCV), October 22-29, 2017, Venice, Italy. New York: IEEE Press, 2017: 2980-2988.

[5] Liu W, Anguelov D, Erhan D, et al. SSD: single shot MultiBox detector[M]//Leibe B, Matas J, Sebe N, et al. Computer vision-ECCV 2016. Lecture notes in computer science. Cham: Springer, 2016, 9905: 21-37.

[6] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 779-788.

[7] Duan Z J, Li S B, Hu J J, et al. Review of deep learning based object detection methods and their mainstream frameworks[J]. Laser & Optoelectronics Progress, 2020, 57(12): 120005.
段仲静, 李少波, 胡建军, 等. 深度学习目标检测方法及其主流框架综述[J]. 激光与光电子学进展, 2020, 57(12): 120005.

[8] Chen L L, Zhang Z D, Peng L, et al. Real-time detection based on improved single shot MultiBox detector[J]. Laser & Optoelectronics Progress, 2019, 56(1): 011002.
陈立里, 张正道, 彭力, 等. 基于改进 SSD 的实时检测方法[J]. 激光与光电子学进展, 2019, 56(1): 011002.

[9] Li C Y, Yao J M, Lin Z X, et al. Object detection method based on improved YOLO lightweight network[J]. Laser & Optoelectronics Progress, 2020, 57(14): 141003.
李成跃, 姚剑敏, 林志贤, 等. 基于改进 YOLO 轻量化网络的目标检测方法[J]. 激光与光电子学进展, 2020, 57(14): 141003.

[10] Cui J H, Zhang Y Z, Wang Z, et al. Light-weight object detection networks for embedded platform[J]. Acta Optica Sinica, 2019, 39(4): 0415006.
崔家华, 张云洲, 王争, 等. 面向嵌入式平台的轻量级目标检测网络[J]. 光学学报, 2019, 39(4): 0415006.

[11] Sandler M, Howard A, Zhu M L, et al. MobileNetV2: inverted residuals and linear bottlenecks[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 4510-4520.

[12] Gao S H, Cheng M M, Zhao K, et al. Res2Net: a new multi-scale backbone architecture[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(2): 652-662.

[13] He K M, Zhang X Y, Ren S Q, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904-1916.

[14] Iandola F N, Han S, Moskewicz M W, et al. SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size[EB/OL]. (2016-11-04)[2020-07-28]. https://arxiv.org/abs/1602.07360.

[15] Zhang X Y, Zhou X Y, Lin M X, et al. ShuffleNet: an extremely efficient convolutional neural network for mobile devices[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT. New York: IEEE Press, 2018: 6848-6856.

[16] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 936-944.

[17] Liu S, Qi L, Qin H F, et al. Path aggregation network for instance segmentation[C]//2018 IEEE/

CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 8759-8768.

[18] Rezatofighi H, Tsoi N, Gwak J, et al. Generalized intersection over union: a metric and a loss for bounding box regression [C] // 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 15-20, 2019, Long Beach, CA, USA. New York: IEEE Press, 2019: 658-666.

[19] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 42 (2): 318-327.

[20] Zhang Z, He T, Zhang H, et al. Bag of freebies for training object detection neural networks [EB/OL]. (2019-04-12) [2020-07-28]. https://arxiv.org/abs/1902.04103.

[21] Fu C Y, Liu W, Ranga A, et al. DSSD: deconvolutional single shot detector [EB/OL]. (2017-01-23) [2020-07-28]. https://arxiv.org/abs/1701.06659.

[22] Dai J F, Li Y, He K M, et al. R-FCN: object detection via region-based fully convolutional networks [EB/OL]. (2016-05-20) [2020-07-28]. https://arxiv.org/abs/1605.06409v2.

[23] Liu S T, Huang D, Wang Y H, et al. Receptive field block net for accurate and fast object detection [M] // Ferrari V, Hebert M, Sminchisescu C, et al. Computer vision-ECCV 2018. Lecture notes in computer science. Cham: Springer, 2018, 11215: 404-419.

[24] Redmon J, Farhadi A. YOLOv3: an incremental improvement [EB/OL]. (2018-04-08) [2020-07-28]. https://arxiv.org/abs/1804.02767.