文章编号　1004-924X(2024)08-1212-15

基于跨层次聚合网络的实时城市街景语义分割

侯志强[1,2]，程敏婕[1,2*]，马素刚[1,2]，屈敏杰[1,2]，杨小宝[1,2]

(1. 西安邮电大学 计算机学院，陕西 西安 710121；
2. 西安邮电大学 陕西省网络数据分析与智能处理重点实验室，陕西 西安 710121)

摘要：随着自动驾驶技术的迅速发展，精确高效的场景理解显得尤为重要。城市街景语义分割旨在准确识别并分割出行人、障碍物、道路和标志物等要素，为自动驾驶技术提供必要的道路信息。然而，当前的语义分割算法在城市街景分割中仍然面临一些挑战，主要表现为不同类别的像素区分不够清晰、对于复杂场景结构的理解不够精准以及对小尺度对象或大尺度结构的分割不准确等问题。为此，本文提出一种基于跨层次聚合网络的实时城市街景语义分割算法。首先，在编码器末端设计了结合跨层次聚合的金字塔池化模块，用于高效提取多尺度上下文信息；其次，在编码器和解码器之间设计了跨层次聚合模块，通过引入通道注意力机制增强信息的表征能力，逐级聚合编码器阶段的特征以充分实现特征复用；最后，在解码器阶段设计了多尺度融合模块，在通道维度聚合全局信息与局部信息，促进深层特征与浅层特征的融合。将所提算法在两个通用的城市街景数据集上进行了验证。在一张 RTX3090 显卡上(TensorRT 测速环境)，本文算法在 Cityscapes 测试集以 294 FPS 的实时性达到 73.0% mIoU 的准确性，在更高分辨率的图像上以 164 FPS 的实时性达到 75.8% mIoU 的准确性；在 CamVid 数据集以 239 FPS 的实时性达到 74.8% mIoU 的准确性。实验结果表明，本文算法在准确性与实时性之间取得了有效平衡，对比其他算法的语义分割性能具有显著提升，为实时城市街景语义分割领域带来了新的突破。

关　键　词：语义分割；卷积神经网络；城市街景；编码器—解码器结构；金字塔池化模块

中图分类号：TP394.1　　**文献标识码**：A　　**doi**：10.37188/OPE.20243208.1212

Real-time urban street view semantic segmentation based on cross-layer aggregation network

HOU Zhiqiang[1,2], CHENG Minjie[1,2*], MA Sugang[1,2], QU Minjie[1,2], YANG Xiaobao[1,2]

(1. *Xi'an University of Posts and Telecommunications, Institute of Computer, Xi'an* 710121, *China*;
2. *Xi'an University of Posts and Telecommunications, Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processing, Xi'an* 710121, *China*)
* *Corresponding author, E-mail: rebu*1999@163.*com*

Abstract: With the rapid development of autonomous driving technology, precise and efficient scene understanding has become increasingly important. Urban street scene semantic segmentation aims to accurately identify and segment elements such as pedestrians, obstacles, roads, and signs, providing necessary road information for autonomous driving technology. However, current semantic segmentation algorithms still face challenges in urban street scene segmentation, mainly manifested in issues such as insufficient dis-

收稿日期：2023-10-21；**修订日期**：2023-12-01.

基金项目：国家自然科学基金资助项目(No. 62072370)；陕西省自然科学基金项目(No. 2023-JC-YB-598)

crimination between different categories of pixels, inaccurate understanding of complex scene structures, and inaccurate segmentation of small-scale objects or large-scale structures. To address these issues, this paper proposed a real-time urban street scene semantic segmentation algorithm based on a cross-layer aggregation network. Firstly, a pyramid pooling module combined with cross-layer aggregation was designed at the end of the encoder to efficiently extract multi-scale context information. Secondly, a cross-layer aggregation module was designed between the encoder and decoder, which enhances the representation ability of information by introducing a channel attention mechanism and gradually aggregates the features of the encoder stage to fully achieve feature reuse. Finally, a multi-scale fusion module was designed in the decoder stage, which aggregates global and local information in the channel dimension to promote the fusion of deep and shallow features. The proposed algorithm was validated on two common urban street scene datasets. On an RTX 3090 graphics card (TensorRT speed measurement environment), the algorithm achieves 73.0% mIoU accuracy on the Cityscapes test set with real-time performance of 294 FPS, and 75.8% mIoU accuracy on higher resolution images with real-time performance of 164 FPS; on the CamVid dataset, it achieves 74.8% mIoU accuracy with real-time performance of 239 FPS. Experimental results show that the proposed algorithm effectively balances accuracy and real-time performance, significantly improving semantic segmentation performance compared to other algorithms, and bringing new breakthroughs to the field of real-time urban street scene semantic segmentation.

Key words: semantic segmentation; convolutional neural network; urban street view; encoder-decoder structure; pyramid pooling module

1　引　言

图像语义分割是城市街景理解领域中一项经典而基础的课题，其目的是在图像中分配像素级标签。准确感知街道场景对于自动驾驶车辆做出正确的判断和规划至关重要，因此语义分割作为场景理解的关键技术成为研究的热点。

随着深度学习算法的兴起，卷积神经网络被应用于图像分割领域，其分割性能大大优于传统的基于手工特征的方法。自全卷积网络[1](Fully Convolutional Network, FCN)提出以来，越来越多的研究者致力于研究性能更好的语义分割模型。PSPNet[2]算法利用金字塔池化模块聚合全局上下文；SFNet[3]算法提出流对齐模块加强特征表示。随着移动设备部署需求的不断增长，实时分割算法[4]受到了越来越多的关注，研究人员设计了许多准确、快速的卷积神经网络(Convolutional Neural Networks, CNN)模型，以满足多种应用的需求。ENet[5]算法通过裁剪网络通道来提高推理速度；ICNet[6]算法限制输入的分辨率，从而降低计算的复杂度；DFANet[7]算法采用轻量级深度可分离卷积减少计算量。这些方法简单而有效。但是，在图像边界部分丢失的空间细节信息降低了分割精度。不同于编码器-解码器结构，Yu等人提出了一种由上下文路径和空间路径组成的双分支分割网络BiSeNet[8]，上下文路径用于增大感受野，空间路径用于保留细节信息。Fan等人提出的STDC-Seg[9]算法中设计的短期密集连接模块(STDC)，具有轻量型和可扩展感受野的特点，以此作为主干网络，能以较低的计算成本增强特征的表达能力。

当前的语义分割算法在城市街景分割领域取得了一些进展，但仍然面临着一系列挑战。首先，街景图像中包含了丰富的语义和结构信息，例如车辆的形状、行人的姿态等，而现有算法未能充分利用这些特征信息，导致对于不同类别的像素区分不够清晰，或者对于复杂场景结构的理解不够精准；其次，街景中的对象存在广泛的尺度差异，例如近距离的行人和车辆与远处的建筑物。多尺度信息未被充分捕获将会导致对小尺度对象或大尺度结构的分割不准确。归其原因，ENet[5]算法对编码器阶段的特征未进行充分的特征复用，造成了有效信息的损失；DFANet[7]算法在网络深层未充分捕获多尺度信息，降低了分

割精度；PSPNet[2]，ICNet[6]，BiSeNet[8]和STDC-Seg[9]算法既对编码器阶段的特征未进行充分的特征复用，又在网络深层未充分捕获多尺度信息。

针对上述问题，本文提出一种跨层次聚合网络（Cross-Layer Aggregation Network，CLANet）应用于实时城市街景语义分割，本文主要工作如下：

（1）在编码器阶段，由于STDC-Seg[9]算法的主干网络具有轻量且高效的特点，因此本文使用STDC-Seg算法的主干网络。为提升网络对于多尺度信息的提取能力，本文设计了结合跨层次聚合的金字塔池化模块（Cross-Layer Aggregation Pyramid Pooling Module，CLA-PPM），从而高效提取多尺度上下文信息。

（2）在编码器与解码器之间，为充分利用编码器阶段的特征信息，本文设计了跨层次聚合模块（Cross-Layer Aggregation Module，CLAM），引入SE[10]通道注意力机制增强信息的表征能力，逐级聚合编码器阶段的特征以充分实现特征复用。

（3）在解码器阶段，设计了多尺度特征融合模块（Multi-Scale Fusion Module，MSFM），在通道维度聚合全局信息与局部信息，使深层特征与浅层特征更好地融合。

在两个通用的城市街景数据集上对所提算法进行了验证。在Cityscapes测试集上，当输入图像大小为512×1 024时，FPS达到294，mIoU达到73.0%；当输入图像大小为768×1 536时，FPS达到164，mIoU达到75.8%；在CamVid数据集上，当输入图像大小为720×960时，FPS达到239，mIoU达到74.8%。与近年多个性能优越的算法进行比较，结果表明，本文算法在准确性和实时性之间取得了有效平衡。

2　相关工作

本节将介绍实时语义分割领域的发展现状，并对本文算法所涉及的金字塔池化模块的相关进展进行介绍。

2.1　实时语义分割

近年来，实时语义分割在实际应用方面迅速发展。为了满足语义分割的实时性需求，研究人员提出了许多方法。ENet[5]算法使用早期下采样策略降低计算成本；ICNet[6]算法设计了一个多分辨率图像级联网络来提高速度；LEDNet[11]算法利用通道缩减和洗牌策略提高速度，利用非对称卷积降低计算成本；DABNet[12]算法构建深度非对称瓶颈模块，通过深度可分离卷积和扩张卷积来提取局部和全局信息；DFANet[7]算法利用子网聚合实现多尺度特征传播，获得足够的感受野并增强模型的学习能力；为了提高效率，BiSeNet[8]算法基于双分支分割网络，分别提取细节特征和语义特征，有效提高了实时网络的准确性；STDC-Seg[9]算法提出具有轻量型和可扩展感受野特点的STDC模块，以此作为主干网络，能以较低的计算成本增强特征的表达能力。本文所提算法的编码器阶段使用了STDC-Seg算法的主干网络，但是仅依赖轻量化主干并不能保证高时效性。因此，本文引入三个关键模块，即CLAM、CLA-PPM和MSFM模块，以在保持高效率的同时提升分割精度。

2.2　金字塔池化模块

提取多尺度的上下文信息对于语义分割尤为重要，有利于提高图像的分割性能。He等人在SPP-net[13]算法中提出了空间金字塔池化模块（SPP）。SPP模块能提取不同尺度的空间信息，提升了模型对于空间布局和物体形变的鲁棒性；受到SPP模块的启发，Chen等人在DeepLabV2[14]算法中提出了空洞空间金字塔池化模块（ASPP），该模块通过具有不同扩张系数的空洞卷积层来构建具有不同感受野的卷积核，以获取多尺度信息；Zhao等人在PSPNet[2]算法中提出金字塔池化模块（PPM），通过多尺度的池化核保留全局信息；Hong等人在DDRNet[15]算法中提出深度聚合金字塔池化模块（DAPPM），通过特征聚合和金字塔池来获取丰富的上下文信息。本文所提算法在DAPPM模块的基础上引入跨层次特征聚合的思想，设计了结合跨层次聚合的金字塔池化模块（CLA-PPM），在减少计算量与参数量的同时保证了分割精度。

3　本文算法

本文提出的跨层次聚合网络（Cross-Layer Aggregation Network，CLANet）分为三个阶段：

编码器阶段、跨层次聚合阶段和解码器阶段，如图1所示。

在编码器阶段，本文算法使用STDC-Seg[9]算法的主干网络，具体如图1中$E_{1/4}$，$E_{1/8}$，$E_{1/16}$和$E_{1/32}$所在的支路。

在跨层次聚合阶段，为充分利用编码器阶段的特征信息，本文设计了跨层次聚合模块（Cross-Layer Aggregation Module，CLAM）以实现特征复用，增强信息的表征能力。此外，本文设计了结合跨层次聚合的金字塔池化模块（Cross-Layer Aggregation Pyramid Pooling Module，CLA-PPM）用于高效提取多尺度特征。

在解码器阶段，逐级将跨层次聚合阶段提供的多尺度信息进行聚合，同时逐渐扩大图像的分辨率，具体如图1中$D_{1/32}$，$D_{1/16}$和$D_{1/8}$所在的支路。此外，本文设计了多尺度融合模块（Multi-Scale Fusion Module，MSFM），在通道维度聚合全局信息与局部信息，使深层特征$D_{1/8}$与浅层特征$E_{1/8}$更好地融合。之后，将融合后的特征M_{out}送入分割头（Seg Head）预测分割结果。

为提升分割精度并保持算法的高时效性，本文在设计模块时致力于选择计算效率高的组件，以最大程度地减小对整体算法的分割速度的影响。在3.1，3.2和3.3节中将详细介绍本文所提出的跨层次聚合模块（CLAM）、结合跨层次聚合的金字塔池化模块（CLA-PPM）以及多尺度融合模块（MSFM）。

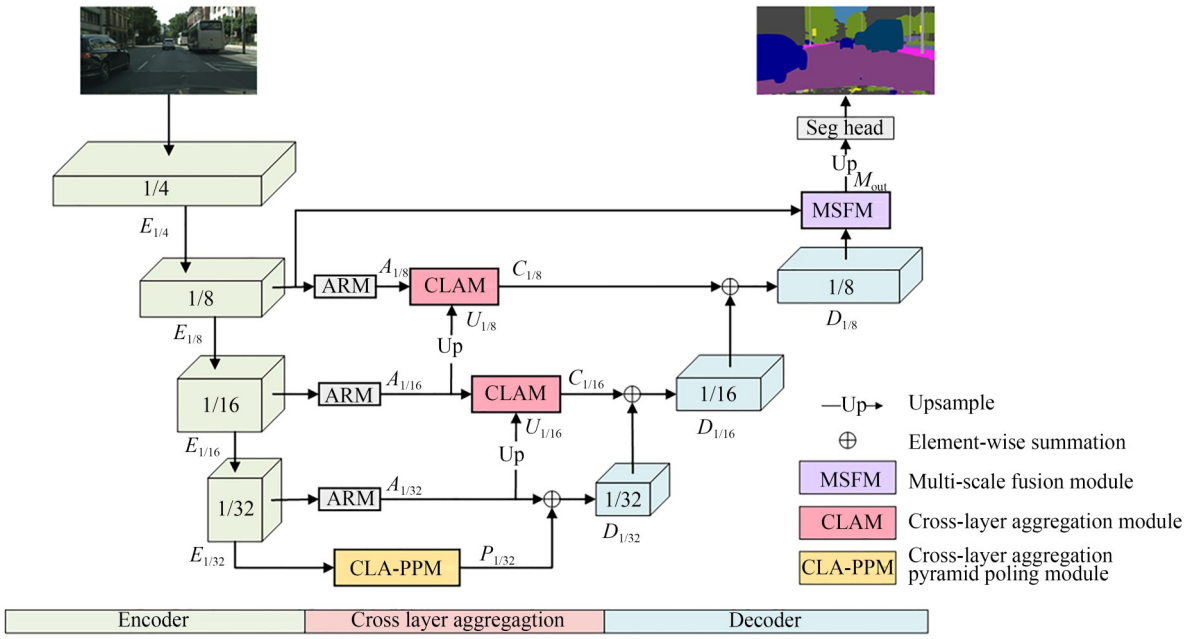


图1　跨层次聚合网络（CLANet）的整体结构

Fig. 1　Overall structure of CLANet (Cross-Layer Aggregation Network, CLANet)

3.1　跨层次聚合模块（CLAM）

为充分利用编码器阶段的特征信息，本文结合SE[10]通道注意力机制设计了跨层次聚合模块（Cross-Layer Aggregation Module，CLAM）。在编码器阶段提取不同层次的特征，将相邻层级的特征相加融合。融合后的特征经过SE通道注意力机制进行通道筛选，强化特征表达的同时减小特征图之间的尺度差异。

本文所提算法中设置了两个跨层次聚合模块（CLAM），具体位置如图1所示。下面以图1中上方的跨层次聚合模块（CLAM）为例进行说明，具体结构如图2所示。

CLAM模块有两个输入，$A_{1/8}$和$U_{1/8}$。在图1中，$A_{1/8}$代表原图1/8大小的特征图（$E_{1/8}$）经过ARM模块得到的输出。ARM（Attention Refinement Module）模块在STDC-Seg算法中用于缩减通道数以减少计算量。$E_{1/8}\in\boldsymbol{R}^{C_2\times H_1\times W_1}$经过ARM模块得到$A_{1/8}\in\boldsymbol{R}^{C_1\times H_1\times W_1}$。$U_{1/8}\in\boldsymbol{R}^{C_1\times H_1\times W_1}$代表

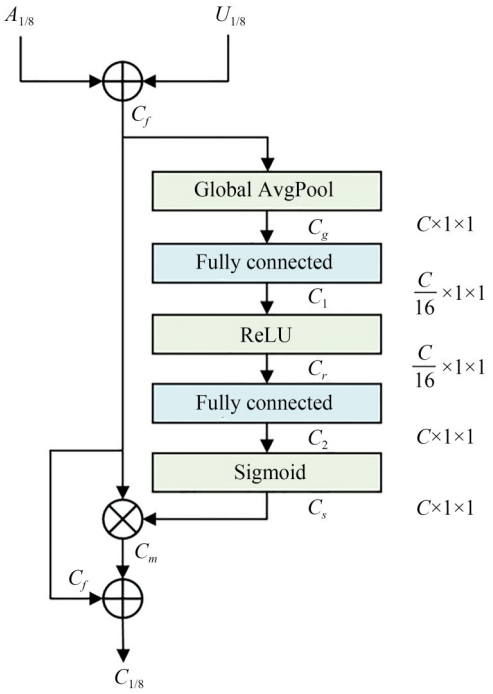


图 2　跨层次聚合模块(CLAM)

Fig. 2　Cross-Layer Aggregation Module

$A_{1/16}\in\boldsymbol{R}^{C_1\times H_2\times W_2}$经过二倍上采样得到的特征图。$A_{1/16}$是由$E_{1/16}\in\boldsymbol{R}^{C_2\times H_2\times W_2}$经过ARM模块缩减通道后所得。

在图2中,首先将来自两个层级的特征$A_{1/8}$和$U_{1/8}$相加融合,得到$C_f\in\boldsymbol{R}^{C_1\times H_1\times W_1}$。如式(1)所示:

$$C_f=A_{1/8}+U_{1/8}. \tag{1}$$

将融合后的特征C_f输入SE通道注意力机制:先将C_f经过全局平均池化操作,得到$C_g\in\boldsymbol{R}^{C_1\times1\times1}$。将$C_g$经过第一个全连接层,同时缩减通道数以减少计算量,得到$C_1\in\boldsymbol{R}^{C_1/16\times1\times1}$。$C_1$通过ReLU激活函数得到$C_r\in\boldsymbol{R}^{C_1/16\times1\times1}$。将$C_r$经过第二个全连接层恢复通道数,得到$C_2\in\boldsymbol{R}^{C_1\times1\times1}$。最后将$C_2$经过Sigmoid激活函数,学习通道之间的关系,得到$C_s\in\boldsymbol{R}^{C_1\times1\times1}$。上述过程表示如下:

$$C_2=FC(ReLU(FC(GAP(C_f)))), \tag{2}$$

$$C_s=\phi(C_2), \tag{3}$$

式中:FC指全连接层,GAP指全局平均池化操作,ϕ指Sigmoid激活函数。

得到通道权重C_s之后,将C_s与融合后的特征C_f相乘,进行特征加权,得到加权后的特征$C_m\in\boldsymbol{R}^{C_1\times H_1\times W_1}$。通过设置残差连接,将$C_m$与$C_f$相加,进行特征复用,重复利用编码器阶段的特征信息,得到$C_{1/8}\in\boldsymbol{R}^{C_1\times H_1\times W_1}$。具体操作如下:

$$C_{1/8}=C_f+C_s\otimes C_f, \tag{4}$$

式中,$\otimes$指逐元素相乘。

3.2　结合跨层次聚合的金字塔池化模块(CLA-PPM)

多尺度特征的提取对于提升分割精度至关重要。Hong等人在DDRNet[15]算法中提出的深度聚合金字塔池化模块(Deep Aggregation Pyramid Pooling Module, DAPPM)在提取多尺度信息方面表现出优越的性能,如图3(a)所示。然而,DAPPM模块将不同尺度的特征图统一进行上采样操作。低分辨率特征图为匹配高分辨率特征图则需进行8倍乃至16倍上采样操作,造成了特征信息的损失。

大多数方法通过逐级融合相邻特征层来实现多尺度特征的融合,但此种做法缺乏特征交互的多样性。为此,DSH-Net[16]算法提出将多尺度特征通过两条并行且不同稀疏度的融合路径进行特征融合。通过并行和分层的方式促进长距离和局部特征交互,从而有效地聚合多尺度特征。同时,多尺度特征之间的语义和分辨率差距在此过程中被弥补。

受到DSH-Net算法的启发,本文将跨层次聚合的思想与DAPPM模块相结合,设计了结合跨层次聚合的金字塔池化模块(Cross-Layer Aggregation Pyramid Pooling Module, CLA-PPM)用于高效提取多尺度上下文信息,避免大倍率上采样操作以减少特征信息的损失,同时实现特征交互的多样性,如图3(b)所示(彩图见期刊电子版)。

DAPPM模块与CLA-PPM模块的开始部分相同,如图3(b)的灰色部分所示。首先,将特征图$E_{1/32}\in\boldsymbol{R}^{C_1\times16\times32}$经过1×1卷积层降维,得到$F_1\in\boldsymbol{R}^{C_2\times16\times32}$与$F_6\in\boldsymbol{R}^{C_2\times16\times32}$。同时,将特征图$E_{1/32}$输入池化内核大小分别为(5,9,17),步长分别为(2,4,8),padding分别为(2,4,8)的平均池化操作(AvgPool),再将上述输出分别经过1×1卷积层降维,得到$F_2\in\boldsymbol{R}^{C_2\times8\times16}$,$F_3\in\boldsymbol{R}^{C_2\times4\times8}$和$F_4\in\boldsymbol{R}^{C_2\times2\times4}$。此外,还将特征图$E_{1/32}$输入全局平均池化操作(Global Average Pooling, GAP)提取

长距离上下文信息，再将输出经过1×1卷积层降维，得到$F_5 \in \boldsymbol{R}^{C_2 \times 1 \times 1}$。

其中，F_1，F_2，F_3，F_4，F_5和F_6的具体计算过程表示如下：

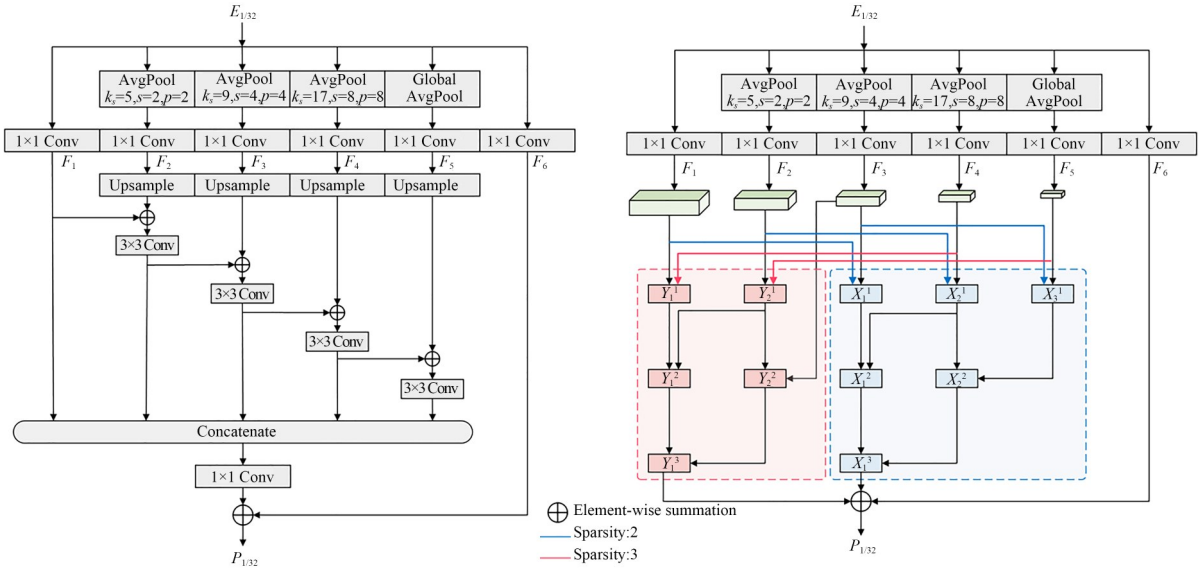
$$F_{1,6} = Conv_{1\times1}(F), \tag{5}$$

$$F_{2,3,4} = Conv_{1\times1}(AvgPool(F)), \tag{6}$$

$$F_5 = Conv_{1\times1}(GAP(F)), \tag{7}$$

式中：$Conv_{1\times1}$指1×1卷积层，$AvgPool$指平均池化操作，GAP指全局平均池化操作。

为丰富多尺度特征$F_1 \in \boldsymbol{R}^{C_2 \times 16 \times 32}$，$F_2 \in \boldsymbol{R}^{C_2 \times 8 \times 16}$，$F_3 \in \boldsymbol{R}^{C_2 \times 4 \times 8}$，$F_4 \in \boldsymbol{R}^{C_2 \times 2 \times 4}$和$F_5 \in \boldsymbol{R}^{C_2 \times 1 \times 1}$之间的特征交互，本文设置了两种稀疏度（Sparsity），分别对应两类特征分组。



Deep aggregation pyramid pooling module(DAPPM)　　Cross-layer aggregation pyramid pooling module(CLA-PPM)

(a) DAPPM模块　　(b) CLA-PPM模块

(a) Represents the DAPPM　　(b) Represents the CLA-PPM

图3　DAPPM模块与CLA-PPM模块的对比图

Fig. 3　Comparison diagram between DAPPM and CLA-PPM

图3(b)中蓝色线段对应稀疏度为2的分组，红色线段对应稀疏度为3的分组。此处稀疏度指图像尺度的间隔。如特征图F_1和F_2的分辨率相差一倍，则称F_1和F_2的稀疏度为1。以此类推，F_1与F_3的稀疏度为2，F_1与F_4的稀疏度为3。

在稀疏度（Sparsity）为2的分组中，将F_1与F_3结合，F_2与F_4结合，F_3与F_5结合。具体结合方式为：低分辨率的特征图经过上采样操作后，与高分辨率特征图相加。相加之后的特征经过SE通道注意力机制，分别得到$X^1{}_1 \in \boldsymbol{R}^{C_2 \times 16 \times 32}$，$X^1{}_2 \in \boldsymbol{R}^{C_2 \times 8 \times 16}$和$X^1{}_3 \in \boldsymbol{R}^{C_2 \times 4 \times 8}$。

在融合操作中，本文结合SE通道注意力机制在通道维度上的优势，突出有效信息并抑制无关信息，从而捕获最具代表性的特征，细化多层次特征的融合。$X^1{}_1$，$X^1{}_2$和$X^1{}_3$的计算过程表示如式(8)：

$$X^1{}_{i-r} = SE(Up(F_i) + F_{i-r})(i \in (5,4,3), r = 2), \tag{8}$$

式中，SE指SE通道注意力机制，Up指双线性插值上采样。

在此基础上，通过同种方式进一步融合$X^1{}_1$，$X^1{}_2$和$X^1{}_3$，以得到$X^2{}_1 \in \boldsymbol{R}^{C_2 \times 16 \times 32}$和$X^2{}_2 \in \boldsymbol{R}^{C_2 \times 8 \times 16}$。具体操作如式(9)：

$$X^2{}_{i-n} = SE(Up(X^1{}_i) + X^1{}_{i-n})(i \in (3,2), n = 1). \tag{9}$$

以此类推，将$X^2{}_1$，$X^2{}_2$以同种方式融合，得到$X^3{}_1 \in \boldsymbol{R}^{C_2 \times 16 \times 32}$。如下式所示：

$$X^3{}_1 = SE(Up(X^2{}_1) + X^2{}_2). \tag{10}$$

为进一步丰富特征交互，引入稀疏度（Sparsity）为3的特征分组。在稀疏度为3的分组中，

将 F_1 与 F_4 结合，F_2 与 F_5 结合。具体结合方式为：低分辨率的特征图经过上采样操作后，与高分辨率特征图相加。相加之后的特征经过SE通道注意力机制，分别得到 $Y^1{}_1\in\boldsymbol{R}^{C_2\times 16\times 32}$ 和 $Y^1{}_2\in\boldsymbol{R}^{C_2\times 8\times 16}$。$Y^1{}_1$ 和 $Y^1{}_2$ 的计算过程表示如式(11)：

$$Y^1{}_{i-m}=SE(Up(F_i)+F_{i-m})(i\in(5,4),m=3). \tag{11}$$

在此基础上，通过同种方式进一步融合 $Y^1{}_1$ 和 $Y^1{}_2$，得到 $Y^2{}_1\in\boldsymbol{R}^{C_2\times 16\times 32}$。同时，将 $Y^1{}_2\in\boldsymbol{R}^{C_2\times 8\times 16}$ 与相邻尺度的特征 $F_3\in\boldsymbol{R}^{C_2\times 4\times 8}$ 进行融合，得到 $Y^2{}_2\in\boldsymbol{R}^{C_2\times 8\times 16}$。上述过程表示如下：

$$Y^2{}_1=SE(Up(Y^1{}_2)+Y^1{}_1), \tag{12}$$

$$Y^2{}_2=SE(Up(F_3)+Y^1{}_2). \tag{13}$$

以此类推，将 $Y^2{}_1$，$Y^2{}_2$ 以同种方式融合，得到 $Y^3{}_1\in\boldsymbol{R}^{C_2\times 16\times 32}$。如式(14)所示：

$$Y^3{}_1=SE(Up(Y^2{}_2)+Y^2{}_1). \tag{14}$$

得到两类特征分组的输出 $X^3{}_1$ 和 $Y^3{}_1$ 之后，将 $X^3{}_1\in\boldsymbol{R}^{C_2\times 16\times 32}$，$Y^3{}_1\in\boldsymbol{R}^{C_2\times 16\times 32}$ 和 $F_6\in\boldsymbol{R}^{C_2\times 16\times 32}$ 相加融合，得到最终的结果。具体操作如式(15)：

$$P_{1/32}=Y^3{}_1+X^3{}_1+F_6. \tag{15}$$

3.3　多尺度融合模块(MSFM)

由于深层特征与浅层特征之间的信息存在差异，仅将它们相加融合或拼接融合都可能会导致特征信息损失。为此，本文提出多尺度融合模块(Multi-Scale Fusion Module，MSFM)，在通道维度聚合全局信息与局部信息，使深层特征与浅层特征更好地融合。多尺度融合模块(MSFM)的具体结构如图4所示。

首先，将浅层特征 $E_{1/8}\in\boldsymbol{R}^{C_2\times H_1\times W_1}$ 与深层特征 $D_{1/8}\in\boldsymbol{R}^{C_1\times H_1\times W_1}$ 进行拼接，得到 $M_C\in\boldsymbol{R}^{C_3\times H_1\times W_1}$。将 M_C 经过1×1卷积层调整通道数，得到 $M\in\boldsymbol{R}^{C_2\times H_1\times W_1}$。上述过程表示如式(16)：

$$M=Conv_{1\times 1}(Concat(E_{1/8},D_{1/8})). \tag{16}$$

将融合后的信息 M 分别通过两条路径。首先将 M 经过全局平均池化操作(Global Average Pooling，GAP)，提取全局上下文信息。将输出经过1×1卷积层降维到原始通道数的1/4大小，以减少计算量。使用ReLU函数进行激活后，经过1×1卷积层还原通道数。最后再经过Sigmoid函数激活，得到注意力权重向量 $S_G\in\boldsymbol{R}^{C_2\times 1\times 1}$。其中，$S_G$ 的计算过程表示如式(17)：

$$S_G=\phi(Conv_{1\times 1}(ReLU(Conv_{1\times 1}(GAP(M))))), \tag{17}$$

式中，ϕ 指Sigmoid激活函数。

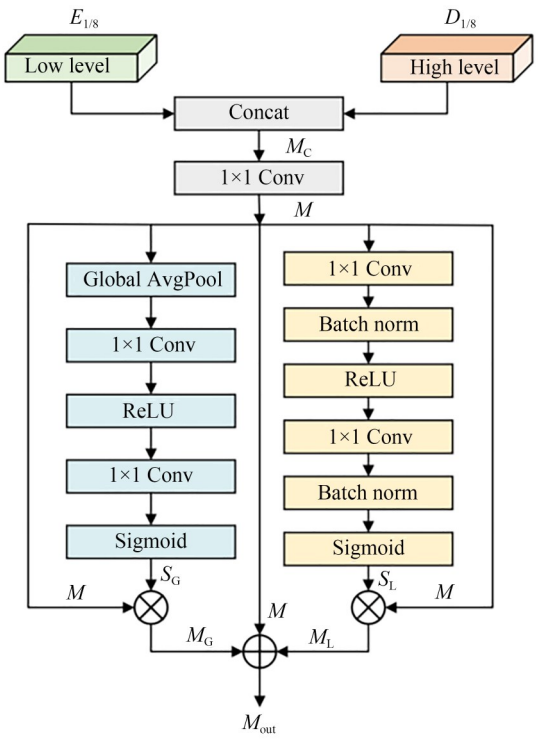


图4　多尺度融合模块(MSFM)

Fig. 4　Multi-Scale Fusion Module, MSFM

与此同时，将 M 经过1×1卷积层降维到原始通道数的1/4大小。经过Batch Norm层、ReLU函数进行激活，再经过1×1卷积层还原通道数。将输出再经过Batch Norm层，Sigmoid函数提取权重信息，得到 $S_L\in\boldsymbol{R}^{C_2\times H_1\times W_1}$。$S_L$ 的计算过程表示如式(18)：

$$S_L=\phi(Conv_{1\times 1}(ReLU(Conv_{1\times 1}(M)))), \tag{18}$$

式中，ϕ 指Sigmoid激活函数。

得到全局权重信息 S_G 与局部权重信息 S_L 后，将 S_G 和 S_L 分别与 M 相乘，进行特征加权，分别得到 $M_G\in\boldsymbol{R}^{C_2\times H_1\times W_1}$ 与 $M_L\in\boldsymbol{R}^{C_2\times H_1\times W_1}$。之后，将 M_G，M_L 和 M 相加，进行特征复用，得到最终的输出 $M_{out}\in\boldsymbol{R}^{C_2\times H_1\times W_1}$。如式(19)所示：

$$M_{out}=M+M\otimes S_G+M\otimes S_L. \tag{19}$$

4　实验结果与分析

为验证本文所提模型的有效性，在Cityscapes和CamVid两个通用的城市街景数据集上训练本文模型，并将其测试精度和推理速度与近年提出的多个性能优越的算法进行比较。

4.1　数据集

Cityscapes[17]是专注于城市街景分析的数据集之一。由5 000张精细标注图像和2万张粗标注图像组成，在本文中仅使用精细标注图像，图像大小为2 048×1 024，每个像素都包含在预定义的19个类中。这些图像被分为训练集、验证集和测试集，分别包含2 975张图像、500张图像和1 525张图像。

CamVid[18]是另一个具有挑战性的城市街景数据集。它是由不同视频序列中提取的701张图像组成，图像大小为960×720，每个像素都包含在预定义的11个类中。这些图像被分为训练集、验证集和测试集，分别包含367张图像、101张图像和233张图像。

4.2　实验环境与参数设置

实验环境：系统环境为Ubuntu 16.04，Python 3.7，PyTorch 1.7.1，GPU为RTX 3090，CUDA版本为11.1。

参数设置：在训练阶段，使用SGD优化器更新网络参数，采用动量（momentum）为0.9的学习策略来降低学习率，权值衰减（weight decay）为$5e^{-4}$。同时采用预热策略和多学习率调度。

训练阶段：Cityscapes数据集进行60 000次迭代训练，初始学习率为0.005，裁剪大小为1 024×512，批量大小（batch size）为24；CamVid数据集进行10 000次迭代训练，初始学习率为0.001，裁剪大小为960×720，批量大小（batch size）为12。损失函数的设置与STDC-Seg[9]算法相同。

推理阶段：采用与训练阶段相同的设备与环境。此外，通过TensorRT 7.2.3.4进行速度测试。分别设置Cityscapes数据集和CamVid数据集的批量大小为1，输入图像的大小分别为512×1 024，768×1 536和720×960。

4.3　消融实验

为验证所提模型的有效性，本文在Cityscapes数据集上进行了CLA-PPM模块的消融实验以及本文所提三个模块对算法整体的消融实验。采用mIoU，FPS（在PyTorch测试环境下）、FLOPs和Parameter指标分别评估了相应的模型，分析、验证各模块对提升算法整体的性能表现，从而证明所提模块的有效性。

4.3.1　结合跨层次聚合的金字塔池化模块（CLA-PPM）的性能

为验证结合跨层次聚合的金字塔池化模块（CLA-PPM）对提取多尺度上下文信息的有效性，在Cityscapes验证集上进行了5组实验，如表1所示。

表1　CLA-PPM在Cityscapes验证集上的消融实验研究

Tab. 1　Ablation study of CLA-PPM on the Cityscapes validation

Baseline	DAPPM	CLA-PPM		FLOPs/G	Params/M	mIoU/%	Speed/FPS
		Sparsity=2	Sparsity=3				
√				97.7	14.2	74.5	96
√	√			98.8	15.53	75.2	83
√		√		98.0	14.68	75.0	89
√			√	97.9	14.52	74.9	90
√		√	√	98.2	15.0	75.3	85

第一组实验即复现STDC1-Seg75的结果，作为本文所提模型的Baseline，第二组至第五组实验均在此基础上进行。第二组实验去除STDC-Seg算法中编码器末端的全局池化操作，引入DAPPM模块提取多尺度上下文信息。第三、第四组实验分别引入本文所提出的CLA-PPM模块中稀疏度（Sparsity）为2的支路与稀疏度（Sparsity）为3的支路。第五组实验引入完整的CLA-PPM模块。

从第一、第二组实验结果可以看出，DAPPM

模块有益于分割精度的提升，mIoU值提高了0.7%；从第一、第三和第四组实验结果可以看出，在CLA-PPM模块中，稀疏度（Sparsity）为2的支路与稀疏度（Sparsity）为3的支路都有益于分割精度的提升，mIoU值分别提高了0.5%，0.4%；从第一、第二和第五组实验结果可以看出，改进后的DAPPM模块，即本文所提出的CLA-PPM模块有益于分割精度的提升。相较于Baseline，mIoU值提高了0.8%。在参数量与计算量方面，CLA-PPM模块的参数量比DAPPM模块少0.53M，计算量少0.6GFLOPs。综上所述，证实了CLA-PPM模块能高效提取上下文信息，有利于提高分割性能。

4.3.2 本文所提三个模块的性能分析

为验证本文所提三个模块在算法中的有效性，设计了8组实验，如表2所示。

表2 本文所提算法在Cityscapes验证集上的消融实验研究

Tab. 2 Ablation study of CLA-Net on the Cityscapes validation

Baseline	CLAM	CLA-PPM	MSFM	FLOPs/G	Params/M	mIoU/%	Speed /FPS
√				97.7	14.2	74.5	96
√	√			103.1	14.5	75.2	86
√		√		98.2	15.0	75.3	85
√			√	98.3	14.22	74.8	93
√	√	√		103.6	15.32	75.7	77
√		√	√	98.8	15.04	75.5	83
√	√		√	103.7	14.53	75.4	84
√	√	√	√	104.3	15.35	76.0	75

第一组实验即复现STDC1-Seg75的结果，作为本文所提模型的Baseline，第二组至第八组实验均在此基础上进行。

从前四组实验结果可以看出，本文所提出的CLAM模块、CLA-PPM模块和MSFM模块都有益于分割精度的提升。第二、第三和第四组实验结果相较于Baseline，mIoU值分别提高了0.7%、0.8%和0.3%，证实了CLAM模块、CLA-PPM模块和MSFM模块的有效性。第五组实验结合CLAM模块和CLA-PPM模块，组成本文中的跨层次聚合阶段。第五组实验结果相较于Baseline，mIoU值提高了1.2%。第六组实验结合CLA-PPM模块和MSFM模块，相较于Baseline，mIoU值提高了1.0%。第七组实验结合CLAM模块和MSFM模块，相较于Baseline，mIoU值提高了0.9%。第八组实验是本文算法的全部组成，相较于Baseline，mIoU值提高了1.5%，从而验证了本文算法的有效性。

4.4 定量分析

本文在Cityscapes和CamVid数据集上对所提算法进行了验证。在通用的城市街景数据集上，本文算法在精确度（mIoU）和速度（FPS）方面取得了良好的平衡。相比于其他引用的算法，这些算法只在特定环境下（PyTorch环境或TensorRT环境）进行了速度测试，而本文算法则同时在PyTorch和TensorRT环境下进行了测试，因此在表3和表4中同时列举了相关算法的性能，进一步凸显了本文算法的优势。在4.4.1和4.4.2节中对这些结果进行了详细的分析。

4.4.1 Cityscapes

如表3所示，本文所提模型与近年实时语义分割的优秀模型对比。其中，CLANet-50和CLANet-75在结构上完全相同，唯一的区别在于输入分辨率。从结果可知，本文方法在速度和准确率之间保持了较好的平衡。针对Cityscapes测试集的数据，具体从以下两方面进行分析。

准确率方面：CLANet-75的mIoU值为75.8%，低于FPANet C[24]的精度。但在PyTorch测试环境下，CLANet-75的速度比FPANet C[24]高44FPS。

速度方面：CLANet-50在TensorRT测试环境下的速度达到294 FPS，在表3中达到最高水平。CLANet-75在TensorRT测试环境下的速度达到164 FPS，高于同等测试分辨率中STDC1-

表 3　本文所提算法在 Cityscapes 数据集上的准确性和速度比较

Tab. 3　Comparison of accuracy and speed of Cityscapes

Method	Reference	Resolution	mIoU/%		#FPS	#FPS
			Val	Test	(PyTorch)	(TensorRT)
GAS[19]	CVPR2020	769×1 537	—	71.8	108.4	—
HMSeg[20]	BMVC2020	768×1 536	—	74.3	83.2	—
DCNet[21]	ICPR2021	512×1 024	—	71.2	142	—
HyperSeg-M[22]	CVPR2021	1 024×2 048	76.2	75.8	36.9	—
RELAXNet[23]	Neurocomputing2022	512×1 024	—	74.8	64	—
FPANet C[24]	APPL INTELL2022	1 024×2 048	—	75.9	31	—
BiAttnNet[25]	SPL2022	512 × 1 024	—	74.7	89.2	—
LETNet[26]	T－ITS2023	512×1 024	—	72.8	150	—
SRDENet[27]	IET COMPUT VIS2023	512×1 024	—	75.4	65	—
BiSeNetV2[28]	IJCV2021	512×1 024	73.4	72.6	—	156
BiSeNetV2-L[28]	IJCV2021	512×1 024	75.8	75.3	—	47.3
FasterSeg[29]	arXiv2019	1 024×2 048	73.1	71.5	—	163.9
STDC1-Seg50[9]	CVPR2021	512×1 024	72.2	71.9	—	250.4
STDC1-Seg75[9]	CVPR2021	768×1 536	74.5	75.3	—	126.7
CPANet-T50[30]	CAC 2022	512 × 1 024	—	72.5	—	234.5
BiSeNetV3-50[31]	Neurocomputing2023	512×1 024	73.4	73.5	—	244.3
CLANet-50(Ours)	—	512×1 024	73.3	73.0	143	294
CLANet-75(Ours)	—	768×1 536	76.0	75.8	75	164

表 4　本文所提算法在 CamVid 数据集上的准确性和速度比较

Tab. 4　Comparison of accuracy and speed of CamVid

Method	Reference	Resolution	mIoU/%	#FPS (PyTorch)	#FPS (TensorRT)
CAS[32]	CVPR2019	720×960	71.8	169	—
GAS[19]	CVPR2020	720×960	72.8	153.1	—
DSANet[33]	ExpertSyst. Appl. 2021	720×960	69.9	75.3	—
FSFNet[34]	IEEE T INSTRUM MEAS2021	720×960	75.1	91	—
RELAXNet[23]	Neurocomputing2022	720×960	71.2	79	—
FPANet B[24]	APPL INTELL2022	720×960	72.9	88	—
LETNet[26]	T－ITS2023	720×960	70.5	200	—
SRDENet[27]	IET COMPUT VIS2023	720×960	74.8	78.3	—
BiSeNetV2[28]	IJCV2021	720×960	72.4	—	124.5
BiSeNetV2-L[28]	IJCV2021	720×960	73.2	—	32.7
STDC1-Seg50[9]	CVPR2021	720×960	73.0	—	197.6
CPANet-T[30]	CAC 2022	720×960	73.9	—	213.9
BiSeNetV3[31]	Neurocomputing2023	720×960	75.1	—	198.4
CLANet(Ours)	—	720×960	74.8	116	239

Seg75[9]的速度。虽然部分算法的分割精度高于本文算法，但本文算法在速度方面更具优势，在准确性和实时性之间保持了良好的平衡。为直观地展示分析结果，本文给出了这些算法在

Cityscapes 测试集上的准确性-速度参数比较的散点图，如图 5 所示。综上所述，本文算法能更精准地分割目标，且综合性能更加优越。

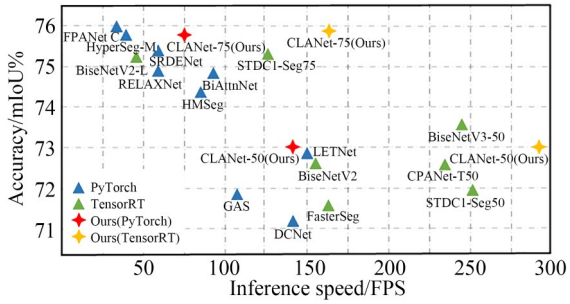


图 5　在 Cityscapes 数据集上的准确性-速度参数比较

Fig. 5　Accuracy-speed comparison on the Cityscapes test set

4.4.2　CamVid

为进一步验证本文算法的有效性，表 4 呈现了本文算法与近年其他优秀的实时语义分割算法在 CamVid 测试集上的对比结果。

准确率方面：本文算法的 mIoU 值达到 74.8%，低于 BiSeNetV3[31] 和 FSFNet[34] 的 mIoU 值。但本文算法的速度高于 BiSeNetV3[31] 和 FSFNet[34]。

速度方面：本文算法在 PyTorch 测试环境下的速度低于 CAS[35]，GAS[22] 和 LETNet[29]，但本文算法的 mIoU 值分别比其高 3.0%，2.0% 和 4.3%。本文算法在 TensorRT 测试环境下的速度达到 239 FPS，在表 4 中达到最高水平。

本文算法在准确性和实时性之间保持了良好的平衡。为直观地展示分析结果，本文给出了这些算法在 CamVid 测试集上的准确性-速度参数比较的散点图，如图 6 所示。从实验结果可以看出，相较于近年来的经典算法，本文算法能更精准地分割目标且综合性能优越。

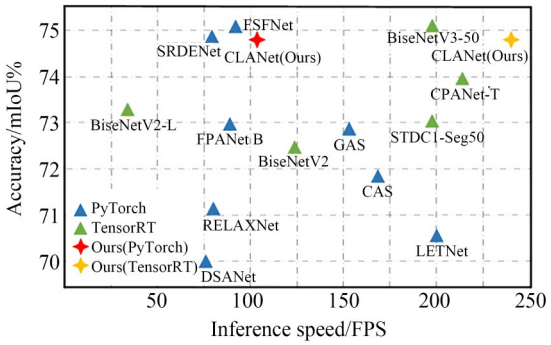
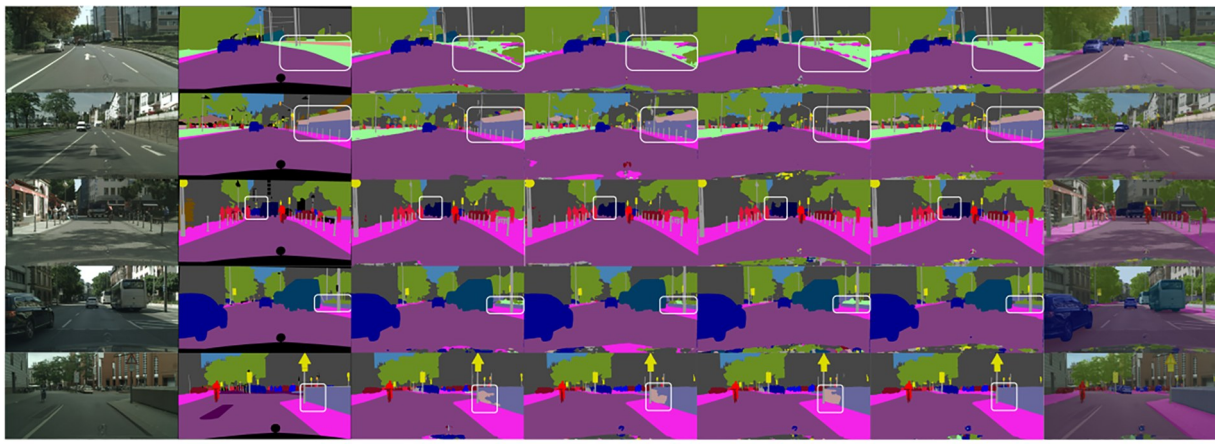


图 6　在 CamVid 数据集上的准确性-速度参数比较

Fig. 6　Accuracy-speed comparison on the CamVid test set

4.5　定性分析

图 7 展示本文算法在 Cityscapes 数据集上和 Baseline 算法、BiSeNetV2 算法以及 HyperSeg 算法的定性分析结果比较；图 8 展示本文算法在 CamVid 数据集上和 Baseline 算法、BiSeNetV2 算法以及 DSANet 算法的定性分析结果比较。



(a) 输入图像 (a) Input image　(b) Ground Truth (b) Ground Truth　(c) HyperSeg-M的输出 (c) Output of HyperSeg-M　(d) BiSeNetV2的输出 (d) Output of BiSeNetV2　(e) STDC1-Seg75的输出 (e) Output of STDC1-Seg75　(f) CLANet-75的输出 (f) Output of CLANet-75　(g) CLANet-75的输出与原图重叠的混合输出 (g) Mixed output

图 7　Cityscapes 数据集的可视化分割结果

Fig. 7　Visual segmentation results of the Cityscapes dataset

4.5.1　Cityscapes

从图 7 所示的可视化结果可以看出，本文算法能准确地判别物体位置并进行分割。

在第一行、第二行和第五行图片中，对于类别频率高、尺度较大的类别如墙面和绿化带，相较于其他算法，本文算法对于对象的分割更准确、完整，减少了语义信息丢失的情况，与标签对比几乎无差异；在第三行图片中，其他算法将车辆像素判别为其他像素的一部分，而本文算法能更好地识别边缘像素；在第四行图片中，本文算法对于草丛的分割更为准确，而其他算法未能较好地判别草丛像素，造成了错分现象。

4.5.2　CamVid

如图 8 所示，整体来看本文算法能够准确识别图像中目标的位置，对于尺度较大的目标，如道路和墙面，分割区域完整，与标签对比几乎无差异；对于尺度较小的目标，如电线杆，其他算法的分割结果较为模糊，杆的主体也有所缺失，而在本文算法的分割结果中，目标的形状明显更加清晰，边界也较为流畅，获得了更精细的分割结果。

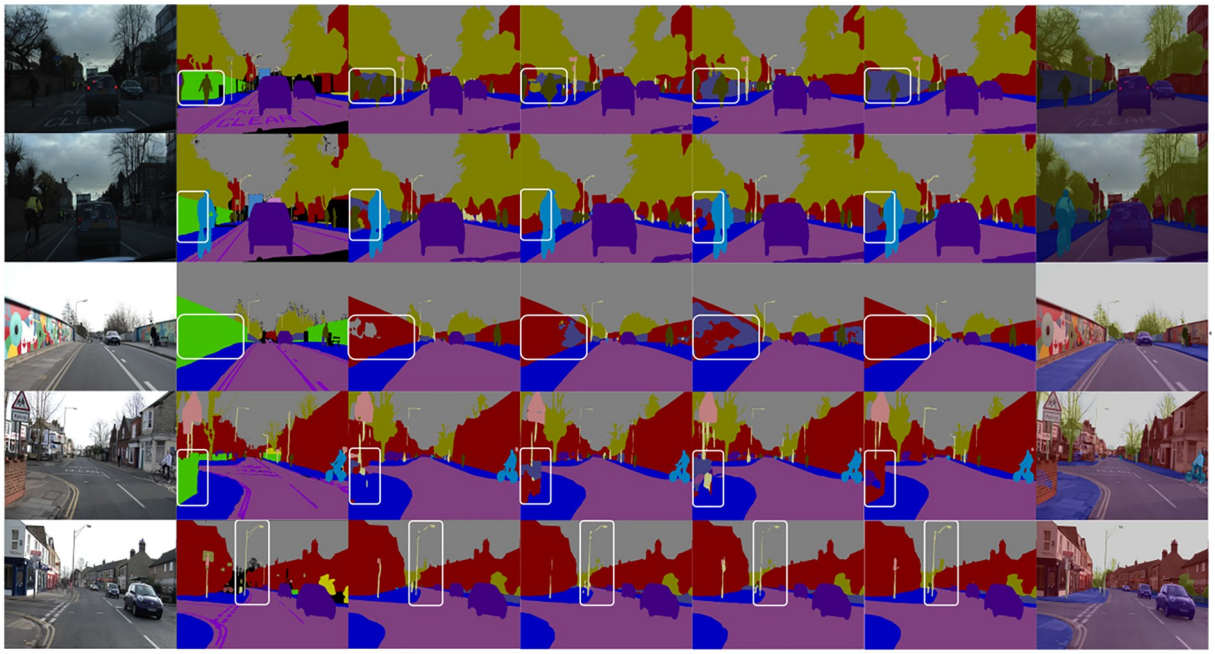


图 8　CamVid 数据集的可视化分割结果

Fig. 8　Visual segmentation results of the CamVid dataset

5　结　论

针对当前的语义分割算法在城市街景分割中面临的一些挑战（包括不同类别的像素区分不够清晰、对于复杂场景结构的理解不够精准以及对小尺度对象或大尺度结构的分割不准确等问题），本文提出了一种基于跨层次聚合网络的实时城市街景语义分割算法。跨层次聚合网络由三个阶段构成：编码器阶段、跨层次聚合阶段和解码器阶段。首先，为加强编码器阶段的特征复用，本文在跨层次聚合阶段设计了跨层次聚合模块（CLAM），结合通道注意力机制增强信息的表征能力，逐级聚合编码器阶段的特征以充分实现特征复用；其次，本文在跨层次聚合阶段设计了一个结合跨层次聚合的金字塔池化模块（CLA-PPM），以高效提取多尺

度上下文信息;最后,本文在解码器阶段设计了多尺度融合模块(MSFM),在通道维度结合全局信息和局部信息,使深层特征与浅层特征更好地融合。本文算法在Cityscapes测试集上以294 FPS的实时性达到73.0% mIoU的准确性,在更高分辨率的图像上以164 FPS的实时性达到75.8% mIoU的准确性;在CamVid数据集以239 FPS的实时性达到74.8% mIoU的准确性,验证了所提出模型的有效性。与其他算法相比,本文提出的语义分割方法在准确性和实时性方面取得了有效的平衡。此外,本文提出的模块具有即插即用的特点,可以应用于其他场景的语义分割任务,为该领域的研究带来新的启示。

参考文献:

[1] LONG J, SHELHAMER E, DARRELL T. Fully convolutional networks for semantic segmentation [C]. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, 3431-3440.

[2] ZHAO H S, SHI J P, QI X J, *et al*. Pyramid scene parsing network [C]. 2017 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*). *Honolulu*, *HI*, *USA*. IEEE, 2017: 6230-6239.

[3] LI X T, YOU A S, ZHU Z, *et al*. *Semantic Flow for Fast and Accurate Scene Parsing*[M]. Computer Vision – ECCV 2020. Cham: Springer International Publishing, 2020: 775-793.

[4] 任凤雷, 杨璐, 周海波, 等. 基于改进BiSeNet的实时图像语义分割[J]. 光学 精密工程, 2023, 31(8): 1217-1227.
REN F L, YANG L, ZHOU H B, *et al*. Real-time semantic segmentation based on improved BiSeNet [J]. *Opt. Precision Eng.*, 2023, 31(8): 1217-1227. (in Chinese)

[5] PASZKE A, CHAURASIA A, KIM S, *et al*. ENet: a deep neural network architecture for real-time semantic segmentation [J]. *ArXiv e-Prints*, 2016: arXiv: 1606.02147.

[6] ZHAO H S, QI X J, SHEN X Y, *et al*. *ICNet for Real-Time Semantic Segmentation on High-Resolution Images*[M]. Computer Vision – ECCV 2018. Cham: Springer International Publishing, 2018: 418-434.

[7] LI H C, XIONG P F, FAN H Q, *et al*. DFANet: deep feature aggregation for real-time semantic segmentation [C]. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*). *Long Beach*, *CA*, *USA*. IEEE, 2019: 9514-9523.

[8] YU C Q, WANG J B, PENG C, *et al*. BiSeNet: bilateral segmentation network for real-time semantic segmentation [M]. *Computer Vision-ECCV 2018*. *Cham*: *Springer International Publishing*, 2018: 334-349.

[9] FAN M Y, LAI S Q, HUANG J S, *et al*. Rethinking BiSeNet for Real-Time semantic segmentation [C]. 2021 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*). *Nashville*, *TN*, *USA*. IEEE, 2021: 9711-9720.

[10] HU J, SHEN L, SUN G. Squeeze-and-excitation networks [C]. 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City*, *UT*, *USA*. IEEE, 2018: 7132-7141.

[11] WANG Y, ZHOU Q, LIU J, *et al*. Lednet: a lightweight encoder-decoder network for real-time semantic segmentation [C]. 2019 *IEEE International Conference on Image Processing* (*ICIP*). *Taipei*, *China*. IEEE, 2019: 1860-1864.

[12] LI G, YUN I, KIM J, *et al*. Dabnet: Depth-Wise Asymmetric Bottleneck for Real-Time Semantic Segmentation [EB/OL]. 2019: *arXiv preprint arXiv*: 1907.11357. https://arxiv.org/abs/1907.11357

[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] 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 and Machine Intelligence, 2018, 40(4): 834-848.

[15] PAN H H, HONG Y D, SUN W C, *et al*. Deep dual-resolution networks for real-time and accurate semantic segmentation of traffic scenes[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2023, 24(3): 3448-3460.

[16] WANG Y P, SHI H, DONG S, *et al*. Dual-path sparse hierarchical network for semantic segmentation of remote sensing images[J]. *IEEE Geoscience and Remote Sensing Letters*, 2021, 19: 8010505.

[17] CORDTS M, OMRAN M, RAMOS S, *et al*. The cityscapes dataset for semantic urban scene understanding[C]. 2016 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*). *Las Vegas*, *NV*, *USA*. IEEE, 2016: 3213-3223.

[18] BROSTOW G J, FAUQUEUR J, CIPOLLA R. Semantic object classes in video: a high-definition ground truth database[J]. *Pattern Recognition Letters*, 2009, 30(2): 88-97.

[19] LIN P W, SUN P, CHENG G L, *et al*. Graph-guided architecture search for real-time semantic segmentation[C]. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*). *Seattle*, *WA*, *USA*. IEEE, 2020: 4202-4211.

[20] LI P, DONG X, YU X, *et al*. When humans meet machines: towards efficient segmentation networks[C]. *The* 31*st British Machine Vision Virtual Conference* (*BMVC*). *September* 7-10, 2020.

[21] LI Y J, LIU Y Z, SUN Q S. Real-time semantic segmentation via region and pixel context network [C]. 2020 25*th International Conference on Pattern Recognition* (*ICPR*). *Milan*, *Italy*. IEEE, 2021: 7043-7049.

[22] NIRKIN Y, WOLF L, HASSNER T. HyperSeg: Patch-wise hypernetwork for real-time semantic segmentation[C]. 2021 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*). *Nashville*, *TN*, *USA*. IEEE, 2021: 4060-4069.

[23] LIU J, XU X Q, SHI Y Q, *et al*. RELAXNet: residual efficient learning and attention expected fusion network for real-time semantic segmentation [J]. *Neurocomputing*, 2022, 474: 115-127.

[24] WU Y, JIANG J Y, HUANG Z M, *et al*. FPANet: feature pyramid aggregation network for real-time semantic segmentation[J]. *Applied Intelligence*, 2022, 52(3): 3319-3336.

[25] LI G L, LI L, ZHANG J W. BiAttnNet: bilateral attention for improving real-time semantic segmentation[J]. *IEEE Signal Processing Letters*, 2021, 29: 46-50.

[26] XU G A, LI J C, GAO G W, *et al*. Lightweight real-time semantic segmentation network with efficient transformer and CNN[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2023, 24(12): 15897-15906.

[27] MI A Z, GAO M M, HUO Z Q, *et al*. Semantics recalibration and detail enhancement network for real-time semantic segmentation[J]. *IET Computer Vision*, 2023, 17(4): 461-472.

[28] YU C Q, GAO C X, WANG J B, *et al*. BiSeNet V2: bilateral network with guided aggregation for real-time semantic segmentation[J]. *International Journal of Computer Vision*, 2021, 129(11): 3051-3068.

[29] CHEN W Y, GONG X Y, LIU X M, *et al*. FasterSeg: searching for faster real-time semantic segmentation[EB/OL]. 2019: *arXiv*: 1912.10917. http://arxiv.org/abs/1912.10917

[30] LIAO Y H, HE L H, DENG Y J, *et al*. Cross guided and pyramid aggregation networks for real-time semantic segmentation[C]. 2022 *China Automation Congress* (*CAC*). *Xiamen*, *China*. IEEE, 2022: 3307-3312.

[31] TSAI T H, TSENG Y W. BiSeNet V3: Bilateral segmentation network with coordinate attention for real-time semantic segmentation[J]. *Neurocomputing*, 2023, 532: 33-42.

[32] ZHANG Y H, QIU Z F, LIU J G, *et al*. Customizable architecture search for semantic segmentation [C]. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*). Long Beach, CA, USA. IEEE, 2019: 11633-11642.

[33] ELHASSAN M A M, HUANG C X, YANG C H, *et al*. DSANet: dilated spatial attention for real-time semantic segmentation in urban street

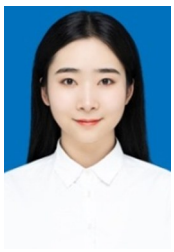
scenes [J]. *Expert Systems with Applications*, 2021, 183: 115090.

[34] PEI Y, SUN B, LI S T. Multifeature selective fusion network for real-time driving scene parsing [J]. *IEEE Transactions on Instrumentation and Measurement*, 2021, 70: 5008412.

作者简介：

侯志强(1973—),男,陕西眉县人,教授,博士生导师,2005年于西安交通大学获得工学博士学位,主要从事图像处理、计算机视觉、无人机应用以及信息融合等方面的研究。E-mail: houzhq@sohu.com

通讯作者：



程敏婕(1999—),女,陕西咸阳人,研究生,2021年于西安工业大学获得学士学位,研究方向为图像语义分割。E-mail: rebu1999@163.com