AGLNet：基于注意力引导的轻量级网络实现自动驾驶图像实时语义分割

Quan Zhou[a,*], Yu Wang[a], Yawen Fan[a], Xiaofu Wu[a], Suofei Zhang[b], Bin Kang[b] and Longin Jan Latecki[c]

文章历史：
2020年1月9日接收
修订版接收日期：2020年7月17日；
接受日期：2020年8月22日
2020年9月2日在线发布

关键词:
机器人视觉
自动驾驶
实时语义分割
卷积神经网络
编码器-解码器网络

摘要

巨大的计算量限制了卷积神经网络（CNNs）在边缘设备上进行图像语义分割的应用，而卷积神经网络对增强现实、机器人和自动驾驶等许多实际应用至关重要。为解决这个问题，本文提出了一种叫做 AGLNet 的注意力引导的轻量级网络，采用编码器-解码器架构实现实时语义分割。具体而言，编码器采用了一种全新的残差模块来提取特征表示，其中使用了通道分割和通道混洗两种新操作，在保持较高分割精度的同时大幅降低计算成本。另一方面，解码器没有使用复杂的空洞卷积和人工设计的复杂结构，而是采用了两种注意力机制来上采样特征以匹配输入分辨率。具体来说，使用分解注意力金字塔模块（FAPM）从高级输出中探索分层空间注意力，同时保持较少的模型参数；为描绘物体形状和边界，采用全局注意力上采样模块（GAUM）作为高级特征的全局引导。综合实验表明，我们的方法在三个自动驾驶数据集 Cityscapes、CamVid 和 Mapillary Vistas 上的速度和精度都达到了先进水平。AGLNet 在这些数据集上分别实现了 71.3%、69.4% 和30.7% 的平均交并比（mIoU），模型参数仅 1.12M，在使用单张GTX 1080Ti GPU 的情况下推理速度分别达到 52 FPS、90 FPS 和 53 FPS。我们的代码已开源，可在 https://github.com/xiaoyufenfei/Efficient-Segmentation-Networks获取。

1. 引言

近年来，构建更深层、更大规模的卷积神经网络（CNN）已成为解决机器人视觉任务的主要趋势，如图像分类[1-3]、目标检测[4-6]和语义分割[7-9]等。为提升视觉数据的表征能力，最精准的卷积神经网络（CNN）通常配有数百乃至数千个卷积层和特征通道，例如ResNet 系列[1,10,11]。尽管性能显著提升，但牺牲了运行时间和推理速度。特别是在增强现实、机器人技术、自动驾驶等现实场景中，通常需要采用计算成本更低、模型规模更小的网络来实现在线估计与决策。因此，那些需要大量计算资源的顶级网络并不适用于计算能力有限的移动平台（如无人机、机器人和智能手机）。这类设备能耗有限，内存有限且计算能力不足。这种局限性在语义分割[8,9,12-14]这类高计算量任务中尤为突出，在机器人视觉领域[15-16]具有重要作用。本文旨在通过将输入图像分割为多个互不重叠的区域，帮助机器人理解周围环境——每个区域都关联着预定义的语义标签，包括天空、道路、建筑物等静态物体，以及人物、车辆、交通信号灯等动态物体。

为适应实际应用需求，设计了多种兼顾分割精度与实现效率的轻量级卷积神经网络。这类网络可大致分为两类：网络压缩[17-20]与卷积分解[21-23]。网络压缩通过压缩预训练模型来减小网络大小，包括哈希[17]、剪枝[18]和量化[19-20]。量化网络[19-20]使用较少的比特来编码模型参数，而不是通过剪枝网络权重[18]来改变网络结构。为进一步消除冗余，另一种轻量级卷积神经网络（CNN）的方法是基于稀疏编码理论[24,25]，该理论使模型权重始终保持稀疏状态，因此在推理计算中仅涉及少量参数。相反，受到将标准卷积分解的卷积分解原理（CFP）[10,22,23,26]的启发，第二类方法侧重于通过减少卷积操作直接训练轻量级网络。例如，分组卷积和深度可分离卷积[2,10]被广泛应用于MobileNet[23,27]和 ShuffleNet 系列[26,28]。ENet[21]将ResNet[1]作为骨干网络实现高效推理。赵等人[29]提出了一种结合高级标签引导的级联网络架构以提升性能。在[13,22,30]中，采用编码器-解码器架构来恢复物体细节，这种方法在保持精度的同时大幅减少了参数数量。尽管这些进展为设计轻量级架构网络在语义分割领域奠定了基础，但在实时机器人视觉语义分割

*通信作者
电子邮箱：quan.zhou@njupt.edu.cn（周全）
https://doi.org/10.1016/j.asoc.2020.106682

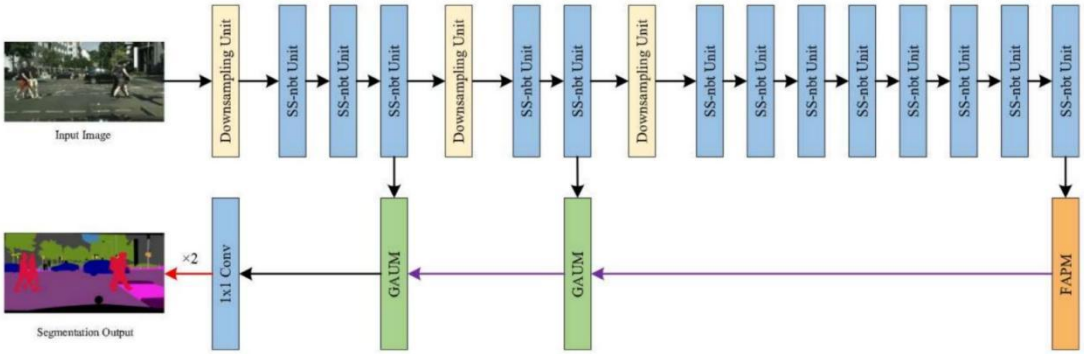

图 1: AGLNet 网络整体架构。编码器采用 SS-nbt 单元提取特征，解码器使用 FAPM 和 GAUM 模块恢复分辨率。

中，如何在有限的计算量下追求最佳精度，仍是亟待解决的开放性课题。

本文旨在整体解决这一权衡问题，同时兼顾精度与运行效率。我们提出了一种新型轻量级网络 AGLNet。AGLNet采用注意力引导的编码器-解码器架构实现实时语义分割。如图1所示，AGLNet 由编码器和解码器网络两部分组成。受 CFP [10,22,23,26]的启发，编码器的核心单元是一个计算高效的残差模块，它在非瓶颈结构（SS-nbt）中采用了分割和混洗操作，该模块利用了恒等映射以及带有通道分割和混洗的一维分解卷积。恒等映射使卷积层能够学习有助于训练的残差函数，而分割与混洗操作在保持与一维分解卷积[22]相似计算量的同时，增强了特征通道间的信息交换。在解码器中，我们摒弃了复杂的空洞卷积[30,31]和人为设计的架构，而是采用两种注意力机制对特征进行上采样以匹配输入图像分辨率。如图1所示，分解注意力金字塔模块（FAPM）通过从低分辨率高阶输出中提取分层空间注意力来提取密集特征，其中分解卷积进一步优化了整个网络的性能。此外，为精确描绘物体的形状与边界，本文采用全局注意力上采样模块（GAUM），利用低层特征的空间信息作为高层语义特征的全局指导，虽然增加了一定的计算量，但性能得到了显著提升。具体而言，模型从低层特征中提取空间注意力，进而对高层特征的各像素位置进行加权。然后，通道注意力被编码以重新加权通道特征，其中丰富的类别信息可用于选择最重要的特征通道。图2 还展示了在 CityScapes 数据集上，最新轻量级语义分割网络在精度与效率方面的性能对比图。可以看出，尽管本文方法在性能方面比 ESPNet [30]和 FPENet [32]要弱，但分割精度提升了10%。与 DABNe t[33]和EDANet [34]相比，本文方法达到了精度与效率之间的最佳平衡。综上所述，AGLNet 的主要优点如下：

- 与先前通常采用对称架构的轻量级网络[13,22，29,30]不同，AGLNet 采用非对称编码器-解码器网络结构。编码器采用 SS-nbt 单元提取下采样特征，在大幅压缩网络规模的同时，依然保持了强大的特征表征能力。随后，解码器在注意力机制的引导下对特征进行上采样，在提升分割精度的同时，依然保持了较高的推理效率。

- AGLNet 的SS-nbt 单元采用分割-变换-合并策略设计残差层，通过一维分解卷积和通道分割与混洗操作，充分利用网络参数并增强特征表征能力，以更低的计算量达到了与大型密集层相当的表征能力。此外，通道混洗是可微的，所以AGLNet 可以以端到端的方式训练。

- FAPM 和GAUM 是两种在解码器中用于提高分割性能的轻量级注意力机制。FAPM 的分层架构扩大了高级特征的感受野，使我们能够收集多尺度上下文。另一方面，GAUM 中编码的空间和通道注意力有效地聚合了低级空间细节并重新校准了特征通道的重要性。

。

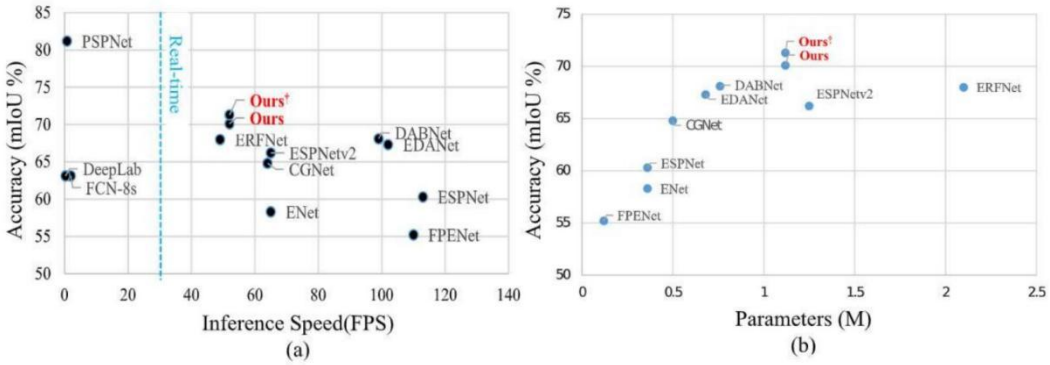

图 2: 与最先进网络在精度与效率权衡方面的比较。从左至右分别为：(a) 分割准确率与帧率的关系；(b) 分割准确率与模型尺寸的关系。"Ours"和"Oursc "分别表示在CityScapes数据集上不使用和使用粗标注数据训练AGLNet 。（最佳效果请以彩色查看）。

• 我们在三个自动驾驶数据集：CityScapes、CamVid和 Mapillary Vistas 上测试了 AGLNet 。综合实验表明，我们的方法在速度和精度方面达到了最优水平。具体来说， AGLNet 在 CityScapes 和 CamVid 测试集上分别实现了 71.3% 和 69.4% 的平均交并比（mIoU）， 在 Mapillary Vistas 验证集上达到 30.7%，模型参数量仅为 1.12M，在三个数据集上使用单张 GTX 1080Ti GPU 的推理速度分别为 52 FPS、90 FPS 和 53 FPS。

本文其余部分结构如下。第2节简要讨论相关工作后，在第3节介绍AGLNet的详细架构。所提出的网络已在CityScapes[35]、CamVid[36]和 Mapillary Vistas[37]三个自动驾驶数据集上进行了评估，实验可在第5节。最后，第6节给出结论性评论和未来工作。

2. 相关工作

在本节中，我们回顾了使用高效 CNN 架构进行机器视觉实时语义分割的相关进展。

2.1. 机器视觉中的实时语义分割

为了帮助机器决策，实时语义分割需要非常快的运行速度以生成高质量的预测结果。ENet[21] 在残差层中设计了一个瓶颈模块来缩小模型。SegNet[13]采用小型网络结构和跳跃连接来达到了较高的运行速度。ICNet[29] 和 ContextNet[38] 利用图像金字塔作为输入来构建级联网络，该级联网络通过融合高级标签引导来提高性能。而 FPENet[39]是使用高效的特征金字塔编码多尺度上下文以实现节省计算成本的。ESPNet[30]采用带有空间金字塔的空洞卷积来提高效率。BiSeNet[40] 使用上下文和空间路径提取高级语义和低级空间特征，并通过将深度计算转移到两个子网络来加速推理速度。在[22,41]中，采用对称的编码器-解码器架构，在保持精度的同时大幅减少参数数量。与这些方法不同，AGLNet采用非对称编码器-解码器架构设计轻量级网络，其中编码器利用分割-变换-合并策略构建残差层，解码器采用FAPM 和 GAUM 两种注意力模块，以实现分割精度和实现效率之间的有效权衡。

2.2. 卷积分解

大多数最先进的高效网络[23,26-28,30,31]使用卷积分解，将标准卷积分解为几个步骤以降低计算复杂度。它们通常将 2D 卷积分解为两个 1D 卷积（例如，将 n × n 分解为 1 × n 和 n × 1），例如分组卷积[3,26,42]、深度可分离卷积[23]及其空洞版本的卷积[30,31]。具体而言，分组卷积[3,26,42]将输入特征通道分成各个小组，每组独立卷积。作为分组卷积的特例，深度可分离卷积[23]在许多高效网络[27,28]中被广泛使用，其中标准卷积分解为深度卷积和 1 × 1点卷积两个步骤。第一步执行轻量级滤波，其中每个输入通道被视为一组，第二步学习所有输入通道之间的线性组合。为从大的有效感受野中学习特征表征，部分轻量级网络[30][31]利用深度可分离空洞卷

积对深度可分离卷积进行扩展。其他代表性网络[22]通过将 2D 卷积（例如，3 × 3）分解为两个1D卷积（例如，1 × 3 和 3 × 1）来减小模型尺寸。与这些高效网络不同，AGLNet 采用 SS-nbt 单元来避免点卷积，节省大量计算量。与仅在输入特征通道的一半上执行卷积的 ShuffleNets[26,28]相比，AGLNet 充分利用所有输入通道以及卷积的多个分支来提高网络表征能力。此外，SS-nbt 单元在保持与 1 维分解卷积相似的计算量的基础上增强了特征通道内的信息交换能力。

2.3. 视觉注意力机制

受语音识别应用[43]的启发，视觉注意力近年来在计算机视觉领域被广泛使用[44-49]。注意力机制可用作全局上下文引导前馈网络以提高性能[46,50]。例如，在[45]中，CNN 的注意力依赖于输入图像的分辨率进行编码。在[46,47]中，通道注意力机制被用于图像识别任务，实现了最佳识别性能。部分注意力网络[40,47,51]通过全局平均池化分支以扩大感受野，并增强密集像素级分类的一致性。EncNet [52]也采用了一种全局池化策略[47]来捕捉高层语义特征，进而预测与这些编码语义相关联缩放因子。FPENet[39]在解码器分支添加了注意力模块。与这些模型不同，AGLNet 采用 FAPM和GAUM 两种注意力模块，分别编码空间注意力与通道注意力，并将二者作为全局上下文来指导分割，从而提升分割性能。

本文的早期版本最早发表于[53]。本文的期刊版本在三个方面扩展了先前发表的版本：(1) 先前版本仍在解码器中使用标准卷积，导致网络计算量大并降低实现效率。相反，我们应用分解的1 D卷积代替 2 D标准卷积，进一步降低模型复杂度。(2) 与仅利用 FAPM 的[53]相比，AGLNet 采用 GAUM 作为全局引导来恢复精确的分辨率细节。(3) 我们开展了更为全面的评估与消融实验，并给出了更充分的对比分析及性能提升结果。

3. AGLNet

在本节中，我们首先介绍编码器中的核心单元SS-nbt，该单元具有分割与混洗操作。然后，提出采用FAPM 和GAUM 两种注意力模块进行语义分割任务。最后，介绍 AGLNet 完整的编码器-解码器网络架构。

3.1. SS-nbt

本文专注于解决残差块中本质上存在的效率限制，残差块用于最新的精确 CNNs 进行图像分类[1,11,26]和语义分割[12,21,22]。为了减少计算量，在残差块中采用分组卷积[11,26]和深度可分离卷积[2,9,23]作为标准步骤。如图3所示，近年来轻量级残差层已有多种成功的设计方案[21,23]。例如，瓶颈模块（图3(a)）源于ResNet[1]的标准残差层，该模块通过减少输入通道数来降低计算资源需求。尽管它通常被用于最先进的网络[19, 21]，但当网络深度增加时性能会急剧下降。另外两个优秀的残差模块是非瓶颈-1D（图3(b)）和ShuffleNet（图3(c)），其中第一个是标准卷积的 1D 版本，而第二个利用瓶颈结构中的点卷积（即 1 × 1 卷积）。然而，[26]提出了相反的观点，认为点卷积占据了大部分计算量，这对轻量级模型非常不利。

为在有限计算量下平衡性能与效率，本文在残差层中引入了通道分割与通道混洗两种简单操作。如图3(d)所示，我们将所提出的该模块命名为分割-打乱-非瓶颈结构（SS-nbt）。受[10,20]的启发，SS-nbt 设计中采用了分割-变换-合并策略，以较低的计算量达到了与大型密集层相当的表征能力。在每个SS-nbt 的开始，输入被分割为两个较低维度的分支，每个分支具有输入的一半通道。为避免点卷积，使用一组分解的 1D 滤波器核（例如，1 × 3, 3 × 1）进行变换，两个分支的卷积输出使用拼接合并，使通道数与输入相同。我们还使用一些分解的1D 空洞卷积来增加感受野。输入特征可以分割为任意数量的分支，而不是两个分支。极端情况下，分支数等于输入特征的通道数，其中每个分支仅包含一个输入特征通道。然而，随着分割分支的增加，会产生对特征内存的重复访问，可能会降低计算效率[27,28]。为促进训练，堆叠的输出通过恒等映射分支与输入相加。最后使用通道混洗[26]来启用两个分割分支之间的信息通信。混洗后，下一个 SS-nbt 单元开始工作。显然，我们的残差模块不仅高效，而且准确。首先，分解卷积包含更少的模型参数，使得每个SS-nbt单元都具有较高的计算效率。与ShuffleNets [26,28]仅对半数输入特征通道进行卷积的网络不同，SS- nbt 的高效率使得我们能够使用更多的特征通道，从而获得更强大的视觉数据表征能力。其次，在每个SS-nbt 单元中，合并的特征通道被随机混洗，然后进入下

一个单元。通道混洗操作可被视为一种特征重用，与视觉数据流一起流向深层网络，这在不显著增加复杂度的情况下扩大了网络容量。

3.2. FAPM

在本节中，我们考虑如何以非常高效的方式为高级特征提供像素级注意力。在实时语义分割场景中，金字塔结构[29,31]已被用于在多个网格尺度上提取特征，其中常采用不同大小的滤波核执行空洞卷积。这种方法尽管在像素级有效扩大了感受野，但空洞卷积却常常会产生网格伪影，这可能损害滤波器响应的局部一致性。此外，这类结构忽视了对高层特征的编码通道注意力与像素级注意力，但它未能考虑卷积的轻量级架构设计。

基于上述观察，我们提出了分解注意力金字塔模块（FAPM），如图4所示。FAPM 由两个注意力部分组成：金字塔特征注意力（PFA）和全局池化注意力（GPA），二者均通过构建像素级注意力以提升性能。为了增加感受野，PFA 采用分层U 形结构[54]，整合来自三个不同金字塔尺度的上下文特征。如图4中绿色箭头所示，为降低计算量，我们首先对每个金字塔尺度采用分解卷积（如尺度1 使用1×7和7×1，尺度2 使用1×5和5×1，尺度3 使用1×3和3×1）并以步长2 进行下采样，以降低特征分辨率。这些下采样后的特征图继续使用步长为1的分解滤波核进行变换处理，以便更好地从每个金字塔尺度中提取上下文线索。由于高级特征图分辨率较小，使用较大的滤波核（例如，1×7 和7×1）不会增加太多计算量。随后，转移的特征通过双线性上采样顺序放大，如图4中红色箭头所示，然后逐步添加到不同尺度的对应部分。最后，将PFA 生成的像素级注意力图，与原始 CNN 特征经1 × 1点卷积变换后的特征图相乘。另一方面，为进一步增强性能，引入GPA 分支以整合全局上下文先验注意力。具体而言，GPA 分支首先利用全局平均池化编码通道注意力，随后通过1 × 1卷积学习输入通道的线性组合。在该分支末端，再次采用双线性上采样以匹配输入特征图的分辨率。

PFA 的分层架构使FAPM 能够捕获多尺度上下文并为卷积特征生成像素级注意力。与文献[9][55]直接堆叠多尺度特征图不同，本文采用分解卷积提取上下文，并与原始卷积特征逐像素相乘，且不会引入太多计算量。

3.3. GAUM

语义分割的编解码网络广泛采用 U 型架构[12,54-56]，通过聚合中间卷积层特征以恢复原始分辨率。部分方法采用简单的解码器结构，仅依赖双线性上采样[8,55]或转置卷积[54]恢复分辨率，忽视了低层空间信息，致使分割结果边界粗糙。另一种方法是[12][56]通过聚合低层与上采样特征以细化物体形状及边界，但导致解码器结构复杂，牺牲了运行速度。其他网络[32][47]则采用全局注意力机制，将高层上下文压缩后嵌入低层特征以提供指导。尽管低层特征有大量的空间细节，高层特征有丰富的语义信息，但二者分辨率与通道维度不同，导致特征融合困难。受[58]的启发，本节介绍一种新的信息融合模块GAUM，它通过编码空间和通道注意力提高视觉数据的表征能力，同时保持较低的计算量。

如图5(a) 所示，GAUM 主要由两部分组成：空间注意力块（SAB）和通道注意力块（CAB）。如图5(a)中紫色箭头所示，首先通过转置卷积来放大高级特征的分辨率。这些上采样的特征图依次乘以 SAB 和 CAB 的输出，前者对各像素位置的滤波响应进行重加权，后者为各特征通道分配不同的权重。最后，加权特征通过元素相加与上采样特征融合。下面，我们分别详细阐述SAB 和CAB 的细节。

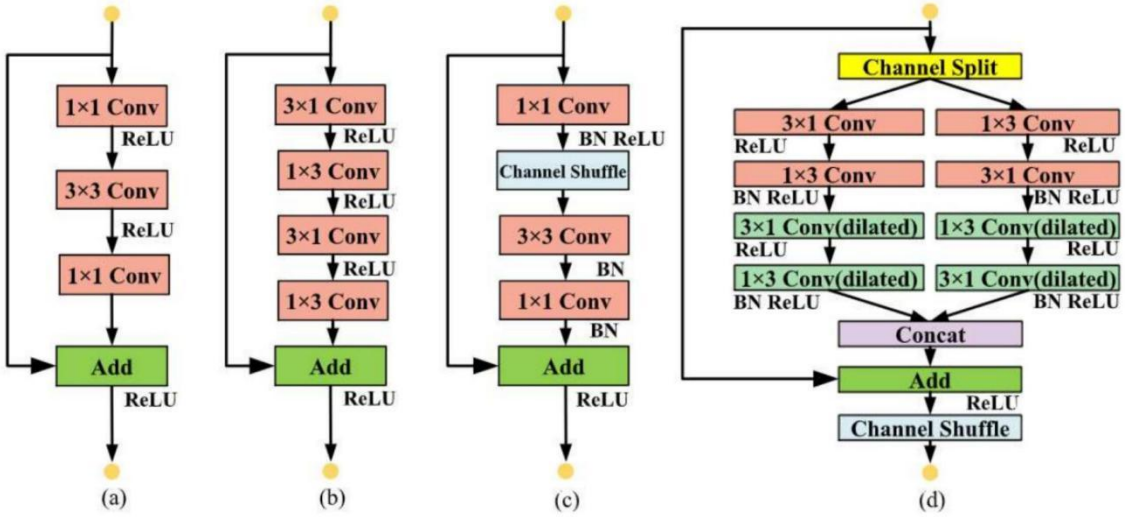

图 3: 不同残差层模块对比。从左至右依次为：(a) bottleneck [21][23], (b) non-bottleneck-1D [22], (c) ShuffleNet [26],and (d) our SS-nbt module.

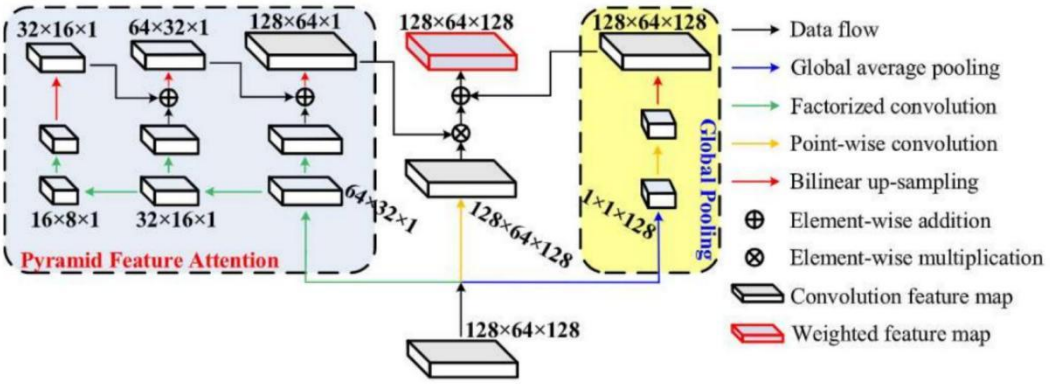

图 4: FAPM 结构:包含金字塔特征注意力（PFA）和全局池化注意力（GPA）两个模块。。

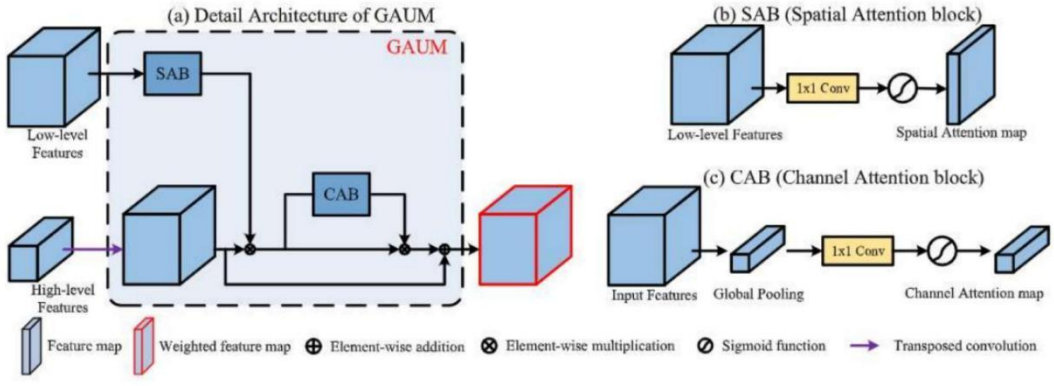

图 5: GAUM 结构。(a) 整体架构；(b) 空间注意力块（SAB）； (c) 通道注意力块（CAB）。

3.3.1. SAB

由于基于类别的分布在不同图像像素上不均匀，SAB 使网络更加关注信息丰富的特征区域。令 x 为输入特征图，f 代表 1 × 1 卷积，∗ 表示卷积操作。然后空间注意力图s 定义为：

$$S=\sigma(X*f) \tag{1}$$

其中 (.) 代表 Sigmoid 函数。变换后，x 的形状从C × H × W 变为 1 × H × W。最后，我们将输入 x 和空间权重图 s 逐元素相乘，得到我们的加权特征图 。

$$X_S=X\otimes S \tag{2}$$

其中 ⊗ 表示逐元素操作。

3.3.2. CAB

我们的通道注意力主要关注不同通道特征的权重差异问题。设 xs(i; j) 表示像素位置 (i; j) 处的特征值。首先对 xs 执行全局平均池化，将各通道的全局空间信息编码为通道描述符：

$$G=\frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W}X_s(i,j) \tag{3}$$

最终，xs 的形状从 C × H × W 变为 1 × 1 × C。与SAB 类似，G 首先直接馈入 1 × 1 卷积层，然后通过Sigmoid 函数，产生通道注意力图 C：

$$C=\sigma(G*f) \tag{4}$$

最终的加权特征图通过将特征图 xs 和注意力图 C 相乘获得：

$$X_{S,C}=X_S\otimes C=X\otimes S\otimes C \tag{5}$$

低层特征生成的空间注意力图标出了每个像素的重要程度，侧重于定位物体并用空间细节细化相应的形状和边界。相反，由高层特征经上采样后生成的压缩通道注意力图反映了每个通道的重要性，其关注全局上下文以提供语义信息。GAUM 融合这两种注意力，将语义概念与空间细节高效嵌入 AGLNet 的各个上采样阶段。

3.4. AGLNet 的网络架构

整个网络架构如图1所示。AGLNet采用轻量级编码器-解码器架构，配备FAPM和GAUM模块。与[13][54]不同，AGLNet 采用非对称顺序架构，其中编码器生成下采样特征图，随后解码器对特征图上采样以匹配输入分辨率。为保留空间信息并减少参数量，总下采样率为8。所提模型的详细结构如表1

表 1
本文的AGLNet的详细架构。输入图像大小为512×1024×3。输出尺寸表示输出特征图的维度，C为类别数。

阶层	层号	操作类型	模式	通道数	输出大小
编码器	1	下采样单元	_	32	256 × 512
	2–4	3× SS-nbt Unit	空洞率 1	32	256 × 512
	5	下采样单元	_	64	128 × 256
	6–7	2× SS-nbt Unit	空洞率 1	64	128 × 256
	8	下采样单元	_	128	64 × 128
	9	SS-nbt Unit	空洞率 1	128	64 × 128
	10	SS-nbt Unit	空洞率 2	128	64 × 128
	11	SS-nbt Unit	空洞率 5	128	64 × 128
	12	SS-nbt Unit	空洞率 9	128	64 × 128
	13	SS-nbt Unit	空洞率 2	128	64 × 128
	14	SS-nbt Unit	空洞率 5	128	64 × 128
	15	SS-nbt Unit	空洞率 9	128	64 × 128
	16	SS-nbt Unit	空洞率 17	128	64 × 128
	17	FAPM	_	128	64 × 128
解码器	18	GAUM	_	64	128 × 256
	19	GAUM	_	32	256 × 512
	20	1 × 1 卷积	步长 1	C	256 × 512
	21	双线性插值	×2	C	512 × 1024

除SS-nbt单元和FAPM外，编码器还包含下采样单元，通过并行堆叠3×3卷积（步长2）和最大池化操作实现。下采样既能帮助更深层的网络收集上下文信息，又能减少计算量。受[10]的启发，本文在编码器中推迟下采样，以保留更多的空间信息，从而有助于提升性能。此外，使用空洞卷积[22][59]使我们的架构具有大的感受野，进一步提高了精度。与使用较大内核尺寸相比，这种技术可以减少计算量和参数数量。对于解码器，两个 GAUM 用于逐步聚合特征并恢复分辨率。然后应用 1 × 1 卷积层将特征通道映射到物体类别数量，并使用 2 × 双线性上采样产生像素级分类器。

4. AGLNet 的端到端训练

在训练 AGLNet 时， 一个主要问题是类别不平衡， 其中每个类别的训练样本数量存在很大差异。一个典型例子是 CityScapes 数据集中的"交通标志"和"道路"类别， 其中第一类物体实例占据非常少的图像区域，而第二类占据大量像素。 因此，我们使用加权交叉熵损失函数以端到端方式训练AGLNet。设 $z_k(x,\theta)$ 为给定网络参数 θ 下，像素 x 属于第 k 类的未归一化对数概率，则 softmax 函数 $p_k(x,\theta)$ 定义为：

$$p_k(x,\theta)=\frac{\exp\{z_k(x,\theta)\}}{\sum_{k'=1}^{K}\exp\{z_{k'}(x,\theta)\}} \tag{6}$$

其中 K 为预定义物体类别的总数。在推理阶段，若第 k 类取得最高预测概率 $k^*=\arg\max_k p_k(x,\theta)$，则将第 k 个语义类别分配给像素 $\mathbf{x}$。

在语义分割任务中，损失函数通常对小批量内的所有像素进行累加。为简化符号，设 N 为批次像素总数，y_i 为像素 $\mathbf{x}_i$ 的真实语义标签，P_{ik} 为其属于类别 k 的预测概率 $p_k(\mathbf{x}_i,\theta)$。训练目标是寻求最优模型参数 θ^*，以最小化加权交叉熵损失函数 $\mathcal{L}(\mathbf{x},\theta)$：

$$\theta^*=\min_\theta \mathcal{L}(\boldsymbol{x},\theta), \tag{7}$$

对于CityScapes数据集[35]，训练样本分布不均往往导致模型偏向于频繁出现的常见类别，而在训练过程中对难以分类的目标改进有限。为解决这个问题，我们利用在线困难样本挖掘（OHEM）方案[60]来定义我们的加权损失函数：

$$\mathcal{L}(\boldsymbol{x},\theta)=-\frac{1}{\sum_{i=1}^{N}\sum_{k=1}^{K}\delta(y_i=k,p_{ik}<\eta)} \times \sum_{i=1}^{N}\sum_{k=1}^{K}\delta(y_i=k,p_{ik}<\eta)\cdot\log(p_{ik}), \tag{8}$$

其中 $\eta\in(0,1]$ 是预定义阈值，$\delta(.)$ 是指示函数，当内部条件成立时等于1，否则等于0。

对于 CamVid 数据集[36]，加权损失函数定义为：

$$\mathcal{L}(\boldsymbol{x},\theta)=-\sum_{i=1}^{N}\sum_{k=1}^{K} w_{ik}\, q_{ik}\log(p_{ik}) \tag{9}$$

其中 $q_{ik}=q(y_i=k|\mathbf{x}_i)$ 表示当像素 $\mathbf{x}_i$ 的语义标签为 k 时的真实分布，w_{ik} 代表权重系数，该系数始终定义为训练数据中第 k 类训练样本计数的倒数[10][21]。

5. 实验

本节呈现了我们在CityScapes[35]和CamVid[36]两个具有挑战性的自动驾驶数据集上的实验结果。此外，还开展了消融研究以深入理解本网络在机器人视觉语义分割任务中的内在机制。

5.1. 数据集

我们在 Mapillary Vistas[37]、CityScapes[35]和CamVid[36]数据集上测试AGLNet，这些数据集是实时语义分割的常用基准。所有数据集都聚焦于城市场景以进行自动驾驶，其中，车辆被视作自主机器人以感知周边环境，具体包括对输入图像中的目标实例进行识别、定位与分割。

表 2
不同方法在Cityscapes测试集上的各类别结果

方法	道路	人行道	建筑物	墙	围杆	电线杆	交通灯	交通标志	植被	地面
ENet [21]]	96.3	74.2	85.0	32.1	33.2	43.4	34.1	44.0	88.6	61.4
ERFNet [22]	97.7	81.0	89.8	42.5	48.0	56.2	59.8	65.3	91.4	68.2
CGNet [41]	95.9	73.9	89.9	43.9	46.0	52.9	55.9	63.8	91.7	68.3
EDANet [34]	97.8	80.6	89.5	42.0	46.0	52.3	59.8	65.0	91.4	68.7
ESPNet [30]	95.7	73.3	86.6	32.8	36.4	47.0	46.9	55.4	89.8	66.0
ESPNet V2 [31]	97.3	78.6	88.8	43.5	42.1	49.3	52.6	60.0	90.5	66.8
FSCNN [61]	97.4	77.8	87.4	39.7	41.8	35.0	39.4	50.5	88.5	63.3
DABNet [33]	97.8	80.7	90.2	47.9	48.1	56.4	61.8	67.0	92.0	69.5
FPENet[32]	96.4	71.7	84.6	27.1	28.8	43.2	39.2	34.4	89.3	61.3
Ours	97.8	81.0	91.0	51.3	50.6	58.3	63.0	68.5	92.3	71.3
Ours[a]	**99.2**	**82.5**	**92.4**	**52.0**	**52.0**	**59.3**	**64.5**	**69.4**	**93.0**	**73.0**
Method	Sky	Ped	Rid	Car	Tru	Bus	Tra	Mot	Bic	mIoU
ENet[21]	90.6	65.5	38.4	90.6	36.9	50.5	48.1	38.8	55.4	58.3
ERFNet [22]	94.2	76.8	57.1	92.8	50.8	60.1	51.8	47.3	61.7	68.0
CGNet [41]	94.1	76.7	54.2	91.3	41.3	55.9	32.8	41.1	60.9	64.8
EDANet [34]]	93.6	75.7	54.3	92.4	40.9	58.7	56.0	50.4	64.0	67.3
ESPNet [30]	92.5	68.5	45.9	89.9	40.0	47.7	40.7	36.4	54.9	60.3
ESPNet V2 [31]	93.3	72.9	53.1	91.8	53.0	65.9	53.2	44.2	59.9	66.2
FSCNN [61]	92.7	65.7	46.4	91.0	**57.0**	**70.3**	**56.5**	40.9	52.6	62.8
DABNet [33]	94.3	80.3	59.2	93.7	46.0	57.1	35.0	50.4	66.8	68.1
FPENet[32]	92.3	68.1	42.7	89.8	29.1	38.9	27.5	29.1	54.5	55.2
Our	94.2	80.1	59.6	93.8	48.4	68.1	42.1	52.4	67.8	70.1
Ours[a]	**95.2**	**81.4**	**60.3**	**95.3**	49.3	69.6	43.5	**53.4**	**69.3**	**71.3**

Mapillary Vistas数据集是一个用于全景分割的大型数据集。它涵盖65个物体类别（28个背景物类别和37个前景物体类别），图像分辨率分布范围较广。该数据集经过了密集标注，其中18K/2K/5K 图像分别用于训练、验证和测试。CityScapes 数据集[35]包含 30 个类别，其中仅 19 个类别（例如，道路、汽车、行人、自行车、天空等）用于语义分割评估。该数据集包含 5000 张高分辨率（2048 × 1024）像素级精细标注图像，其中2975张图像用于训练，500张图像用于训练、1525 张图像用于测试。它还包含另一组近 20,000 张粗略标注的图像。而CamVid[36]是一个较小的数据集，仅涉及 11 个物体类别，701 张图像。所有图像都从5 个分辨率为960×720的视频中收集，其中分割原则为367 张用于训练，101 张用于验证，233 张用于测试。为公平比较，我们将原始图像尺寸下采样到 1024×512和480×360，分别作为两个数据集的输入分辨率。

5.2. 基线

为了展示我们方法的优势，我们选取了 9 个最先进的模型作为基线进行比较，包括ENet [21]、ERFNet [22]、ESPNet [30]、ESPNet V2 [31]、CGNet [41]、EDANet [34]、FSCNNNet [61]、FPENet [32]和DABNet[33]。部分基线模型的实验结果采用作者提

供的默认参数设置复现得到，其余结果则直接引用已发表的文献。所有基线模型均采用平均交并比（mIoU）[22][61] 类别分数进行评估。该指标的计算方式为：先求取模型预测结果与真实标注之间的交集与并集的比值，再对所有数据集中的所有语义类别取平均值。

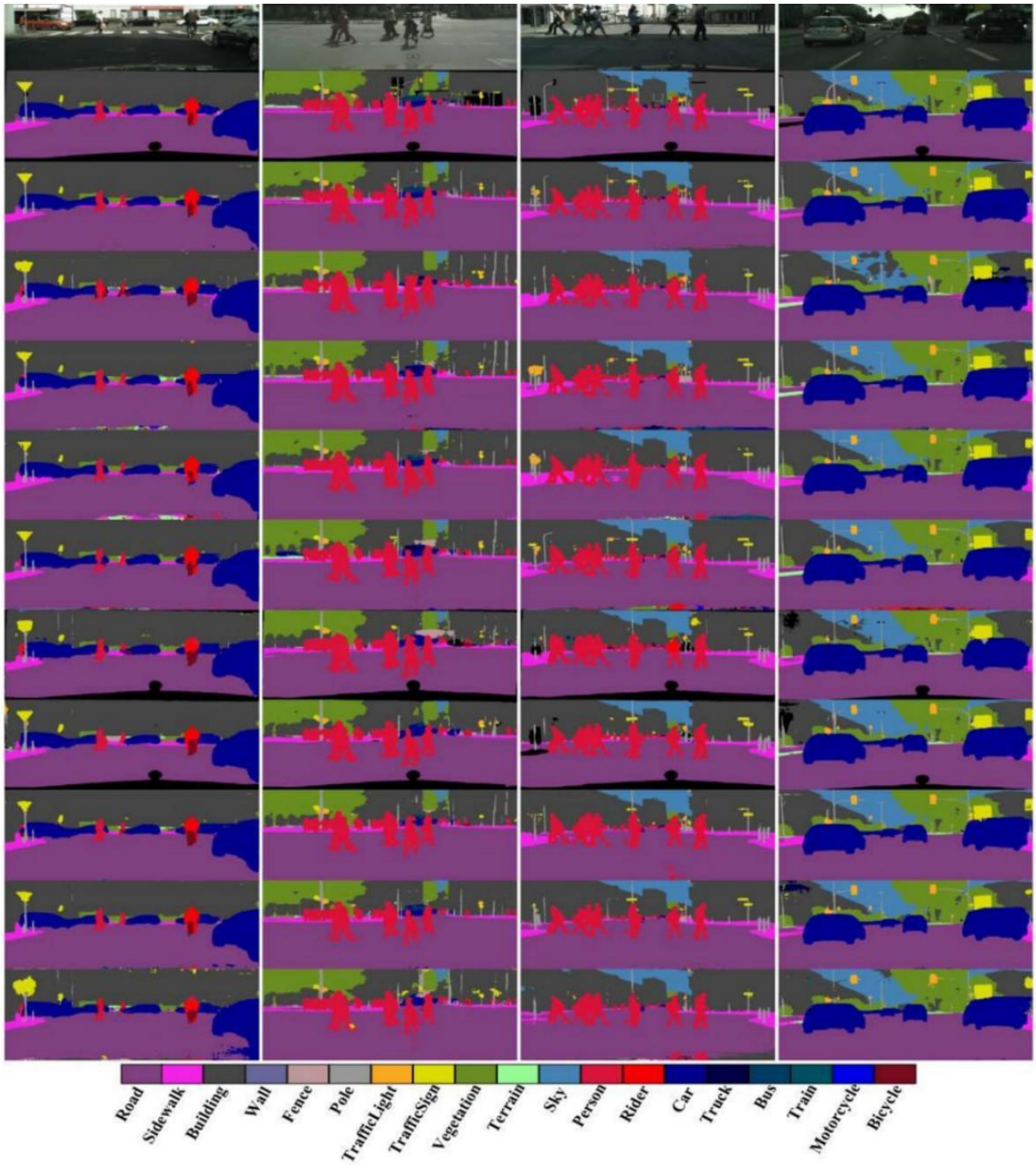

图 6: 在Cityscapes验证集上的视觉对比。从上到下依次为：输入图像、真实标签以及AGLNet 、 ENet [21] 、 ERFNet [22]、 CGNet [41] 、EDANet [34] 、ESPNet [30]、ESPNetv2 [31]、FSCNNet [61]、DABNet [33]和FPENet [32]的分割输出结果。（建议彩色观看）

5.3. 实现细节

AGLNet 在配备单张 GTX 1080Ti GPU 的 Dell 工作站上实现，使用 Adam 优化器[61]对 CityScapes 数据集进行端到端方式训练，对于 CamVid 数据集使用 Rectified Adam (RAdam)[64] 结合LookAhead[65]的方式训练。对于两个数据集，为了充分利用 GPU 显存，我们将训练批量大小设为 8。初始学习率设为$5\times e^{-4}$和$3\times e^{-2}$。在我们的训练过程中采用"poly"学习率策略[14]：学习率通过$(1-\frac{iter_num}{max_iter})^{power}$更新，power 为 0.9，动量和权重衰减分别设为 0.9、10^{-4}，最大训练 epoch 数对于两个数据集分别设为 500 和1000。在数据增强上，我们在训练期间对输入图像采用随机水平翻转、左右翻转和 0.5 到 2 之间的随机缩放。最后，我们随机裁剪图像为固定尺寸进行训练。两个数据集的所有图像都归一化为零均值和单位方差。

表3
AGLNet在Cityscapes测试集上的速度与准确率比较。除与轻量级基线方法对比外，我们还与一些重量级模型进行了比较，包括FCN 、PSPNet 和DeepLab 。

方法	输入尺寸	额外数据	参数量	计算量	帧率	mIoU (%)
SegNet [13]	640 × 360	ImN	29.5M	286G	16.7	57
FCN-8S [8]	512 × 1024	no	_	136.2G	2	63.1
DeepLab [9]	512 × 1024	ImN	262.1M	457.8G	0.25	63.1
RefineNet [12]]	512 × 1024	ImN	118.1M	526G	9.1	73.6
OCNet [62]	512 × 1024	ImN	62.6M	549G	8.7	80.1
PSPNet [55]	713 × 713	ImN + Coa.	250.8M	412.2G	0.78	81.2
ENet [21]	512 × 1024	no	0.36M	4.4G	65	58.3
ERFNet [22]	512 × 1024	no	2.1M	26.86G	49	68.0
CGNet [41]	512 × 1024	no	0.5M	7.01G	64	64.8
EDANet [34]	512 × 1024	no	0.68M	8.95G	102	67.3
ESPNet [30]	512 × 1024	no	0.36M	4.7G	113	60.3
ESPNetv2 [31]	512 × 1024	ImN	1.25M	5.85G	65	66.2
FSCNN [61]	512 × 1024	Coa.	1.14M	1.76G	**230**	62.8
DABNet [33]	512 × 1024	no	0.76M	10.46	99	68.1
FPENet [32]	512 × 1024	no	**0.12*M***	**1.58*G***	110	55.2
Ours	512 × 1024	no	1.12M	13.88G	52	70.1
Ours+	512 × 1024	Coa.	1.12M	13.88G	52	**71.3**

"ImN"和"Coa."分别表示使用ImageNet数据集或Cityscapes粗标注集进行预训练的模型。"—"表示相应方法未提供结果。我们复现了部分模型，并在与AGLNet相同的设置下评估其速度，以确保比较的公平性。

5.4. CityScapes 上的评估结果

为公平比较，所有基线都在相同的硬件平台上使用单张 NVIDIA GTX 1080Ti GPU 进行实验。表2和表3比较了 AGLNet 与选定的最先进网络的相关数据，结果表明，AGLNet 在分割准确率方面优于这些基线方法，同时仍保持了实时运行效率。在所有这些方法中，本文方法的模型大小仅有1.12M，在不使用额外训练数据的情况下，达到了52 帧/秒的推理速度和70.1%的mIoU。仅使用额外的粗略标注训练数据，分割精度可再提高 1.2%，达到71.3%的mIoU。从表2可以看出，19 个物体类别中的16 个获得了最佳mIoU 值，特别是对于某些类别，比排名第二的方法取得了显著改进（例如，"墙"提高 8.1%，"摩托车"提高2.1%）。关于效率，AGLNet的尺寸几乎是ERFNet[29] 的一半，但速度比ERFNet快。其他轻量级基线比 AGLNet 更快，但牺牲了分割精度。例如，FSCNN 虽然达到了最高的推理速度，但其分割准确率比 AGLNet 低了 8.5%。我们还与一些重量级模型进行了比较，结果列于表3。结果表明，本文方法性能比无法达到实时推理的[8][9]更优。图3展示了在CityScapes验证集上的一些定性结果。结果表明，与基线方法相比，AGLNet不仅能够正确分类不同尺度的物体，而且对所有类别都产生了一致的定性结果。

5.5. CamVid 上的评估结果

我们还在 CamVid [51]数据集上对 AGLNet 进行了评估，结果列于表4和表5。与选定的最先进基线。相比，AGLNet 在运行速度和分割精度方面表现出优越的性能。由表4可见，除"天空"类别外，AGLNet在其余类别中均达到最佳性能。值得注意的是，由于输入图像分辨率较低（Cityscapes 为1024 × 512，CamVid 为480 × 360），AGLNet 在 CamVid 数据集上的运行速度更快（90 FPS vs 52 FPS）。图7 展示了一些分割输出的视觉示例。

。

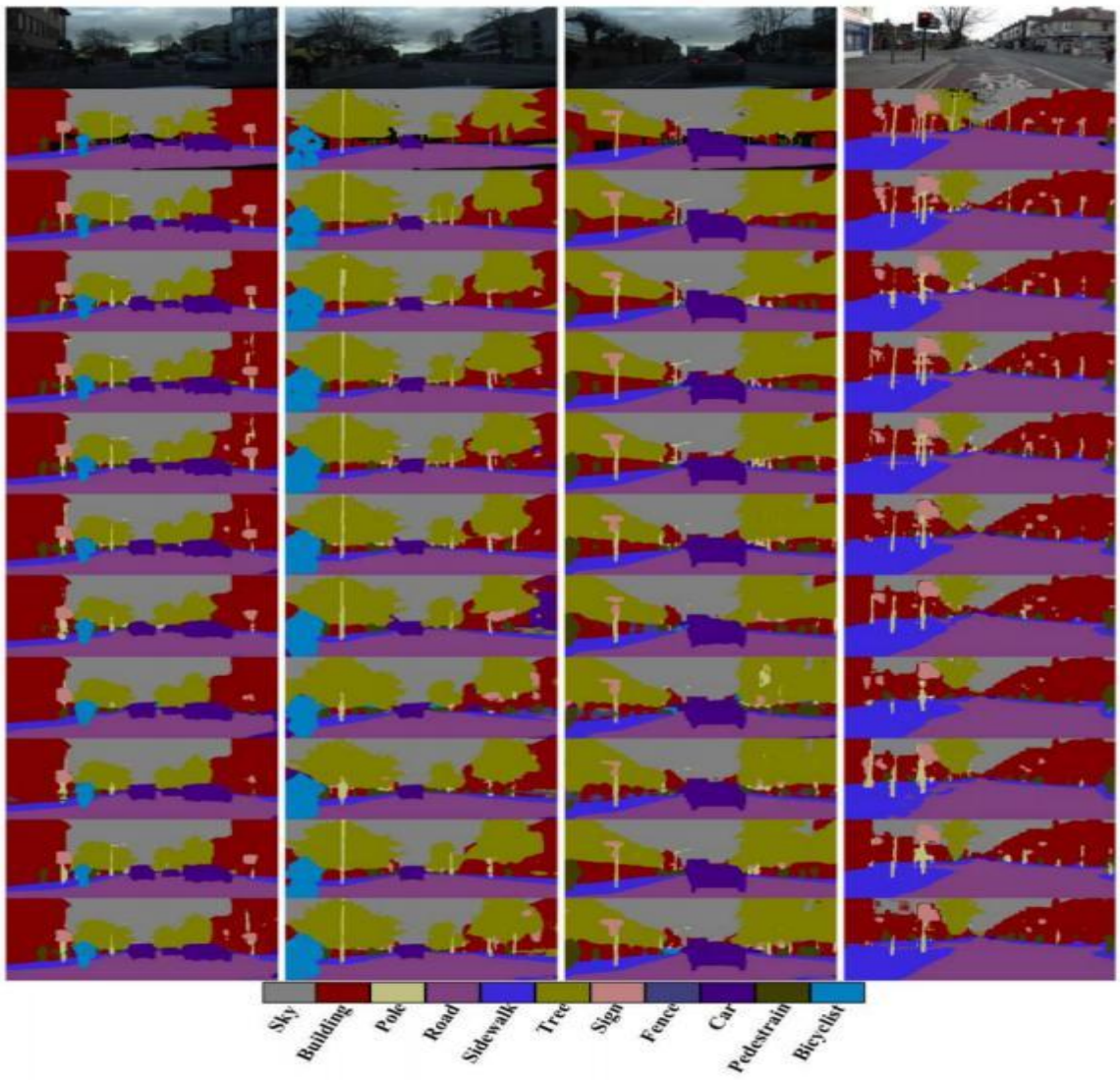

图 **7:** 在CamVid验证集上的视觉对比。从上到下依次为：输入图像 、真实标签以及AGLNet 、ENet [21] 、ERFNet [22]、CGNet [41] 、EDANet [34] 、ESPNet [30] 、ESPNetv2[31]、FSCNNet [61]、DABNet [33]和FPENet [32]的分割输出结果。（建议彩色观看）

表4
CamVid测试数据集上各独立类别的准确率结果。

方法	天空	建筑物	电线杆	道路	人行道	树	交通标志	围栏	车	行人	骑行者	mIoU (%)
ENet [21]	91.2	74.9	23.4	92.1	73.7	68.1	30.1	20.9	77.3	41.1	45.8	58.1
ERFNet [22]	92.0	81.3	37.8	95.0	81.1	75.0	45.0	36.2	84.3	58.3	58.2	67.7
CGNet [41]	90.8	79.8	28.1	95.3	81.9	73.2	41.6	32.9	81.3	52.9	53.9	64.7
EDANet [34]	89.8	79.4	24.3	94.0	81.0	71.1	37.3	31.4	76.9	51.1	53.5	62.7
ESPNet [30]	**92.0**	75.0	25.0	91.5	73.8	68.4	29.5	23.7	74.5	42.4	45.2	58.2
ESPNetv2 [31]	91.0	71.0	18.1	90.1	67.2	61.3	20.0	21.1	69.7	28.8	33.4	52.0
FSCNN [61]	90.2	74.3	15.0	91.7	72.6	67.9	28.9	17.4	70.1	31.9	35.6	54.2
DABNet [33]	91.1	81.0	29.4	93.8	78.7	74.1	43.0	37.2	81.7	56.2	56.5	65.7
FPENet [32]]	91.0	76.3	31.0	93.8	78.3	68.8	32.1	25.1	77.7	45.6	45.6	60.5
Ours	91.8	**82.6**	**39.0**	**95.4**	**83.1**	**76.1**	**45.3**	**39.5**	**87.0**	**61.5**	**62.7**	**69.4**

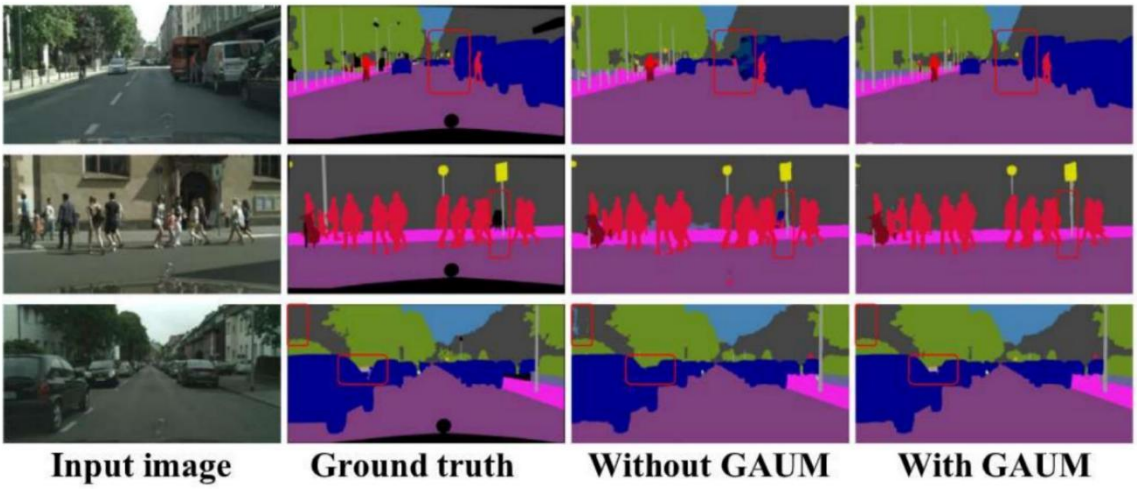

图 8: FAPM在Cityscapes验证集上的可视化结果（最佳效果为彩色显示）。

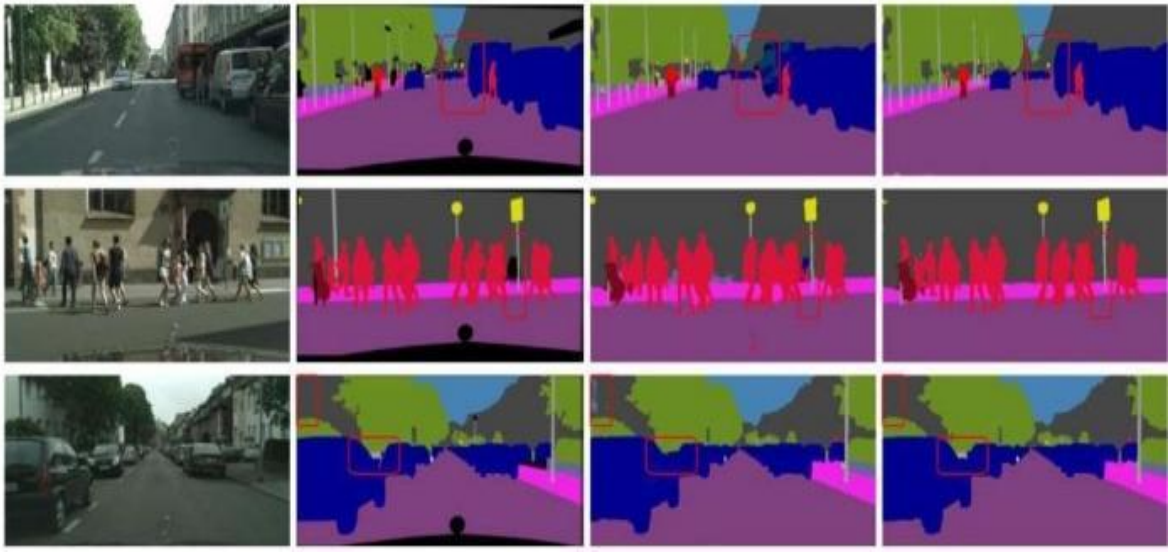

图 9. GAUM在Cityscapes验证集上的可视化结果（最佳效果为彩色显示）。

5.6. Mapillary Vistas 上的评估结果

随后，我们在表6中验证本文方法在 Mapillary Vistas 数据集上扩展类别数量时的表现。为验证本文方法在该数据集上的性能，我们选择DABNet [33]和FPENet[32]作为基线。与 Cityscapes 和CamVid 数据集相比，由于Mapillary Vistas 数据集类别数量较多，本文方法仅获得 30.7%的 mIoU。即便如此，表6仍表明本文方法在 mIoU 方面优于 FPENet [32]（30.7%对比28.33%）。尽管DABNet [33]的运行速度几乎是本文网络的两倍，但本文方法在分割准确率上实现了 1.1%的提升。

表5

与CamVid测试数据集上的最先进网络进行比较。

方法	参数量(M)	FPS	mIoU (%)
ENet [21]	0.36	98.8	58.1
ERFNet [22]	2.10	139.1	67.7
CGNet [41]	0.50	99.1	64.7
EDANet [34]	0.68	175.1	62.7
ESPNet[30]	0.36	190.3	58.2
ESPNetv2 [31]	1.25	118.5	52.0
FSCNN [61]	1.14	**245.1**	54.2
DABNet [33]	0.76	164.5	65.7
FPENet [32]	**0.12**	116.5	60.5
Ours	1.12	90.1	**69.4**

表6

在Mapillary Vistas验证集上与最先进方法的mIOU比较。

方法	输入大小	额外数据	参数量	FLOPs	FPS	mIoU (%)
FPENet [32]	1024 × 2048	no	0.76M	20.9	**103**	28.33
DABNet [33]	1024 × 2048	no	0.12M	**3.10**	75	29.60
Ours	1024 × 2048	no	1.12M	24.12	53	**30.70**

表 7

AGLNet在Cityscapes验证集上的消融实验结果

模型	FAPM	GAUM	训练集	验证集	mIoU (%)	参数量(M)
AGLNet			✓		66.12	0.91
AGLNet	✓		✓		67.62	0.95
AGLNet		✓	✓		69.19	1.08
AGLNet	✓	✓	✓		69.39	1.12
AGLNet[a]	✓	✓	✓	✓	74.50	1.12

5.7. 消融研究

为了验证所提 AGLNet 中两个注意力模块的有效性，我们在Cityscapes 验证集上进行了消融实验，分别将 FAPM 和 GAUM 加入系统，并将二者结合使用。表7报告了各组分及其组合在 mIoU 方面的贡献。研究发现，引入更多注意力模块可提升性能。与未采用注意力模块的基线模型相比, 仅使用 FAPM 的模型达到 67.62%的mIoU，性能提升了1.5%。而单独使用GAUM 则比基线高出3.07%，达到69.19%的分割准确率。这是由于，与FAPM 相比，GAUM 充分利用了高层特征的语义信息和低层特征的空间细节作为交互指导，从而提升了性能。另一个有趣的现象是，使用 GAUM 的模型尺寸比使用FAPM 稍大（例如，0.95M vs. 1.08M）。这可能是因为我们在AGLNet 中使用了两个GAUM 单元。当同时使用两种注意力模块时，AGLNet 达到了最高的分割准确率，提升至 69.39%。此外，我们还使用验证集来训练同时配备FAPM 和GAUM 的AGLNet，达到了 74.5%的mIoU，这表明更多的训练数据有助于进一步提升性能。 一些视觉示例如图8和图9所示，其结果与表7一致。具体而言，FAPM的效果可在图8中直观观察到：部分细节和物体边界更加清晰（例如第一和第三个示例中的"建筑物"和"人行道"），且一些被遗漏的微小物体也得到了正确分类（例如第二个示例中的"自行车"和"交通标志"）。FAPM通过捕获多尺度上下文信息增强了判别能力。与此同时，图9表明，采用GAUM后，部分被错误分类的类别如今得到了正确分类，例如第一个示例中的"汽车"和第三个示例中的"树木"。语义一致性得到了提高。

6. 结论与未来工作

本文提出了一种 AGLNet 模型，设计了轻量级编码器-解码器网络用于自动驾驶图像的实时语义分割。编码器在残差层中采用通道分割与打乱操作，以特征复用的方式增强信息交流；解码器则采用FAPM 和GAUM 两个注意力模块，前者利用空间金字塔架构在不显著增加计算开销的前提下扩大感受野，后者通过高低层特征交互指导来提升性能。整个网络采用端到端方式训练。为评估本文方法，在 Cityscapes 和 CamVid 两个主流自动驾驶数据集上进行了实验，结果表明AGLNet 在分割准确率与实现效率之间达到了最佳平衡。未来，我们将致力于模型参数量化，以进一步提升实时语义分割的运行速度。

作者贡献声明

周全：概念化、方法论、撰写初稿。王宇：软件、验证、调查、数据整理。范雅文：审阅与编辑、可视化。吴小福：审阅与编辑、监督。张索飞：审阅与编辑、项目管理。康斌：审阅与编辑。Longin Jan Latecki：审阅与编辑。

利益冲突声明

作者声明他们没有已知的可能影响本文报告工作的竞争性经济利益或个人关系。

致谢

作者要感谢副主编和所有匿名审稿人的宝贵意见和深刻建议。本工作部分由国家自然科学基金（61876093、61801242、61671253）、江苏省自然科学基金（BK20181393）、美国国家科学基金会（IIS-1302164）以及国家留学基金委（201908320072）联合资助。

参考文献

[1] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recog- nition, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[2] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.

[3] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Annual Conference on Neural Information Processing Systems, 2012, pp. 1097–1105.

[4] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in: Annual Conference on Neural Information Processing Systems, 2015, pp. 91–99.

[5] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You nly look once: unified, real-time object detection, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.

[6] R. Girshick, Fast R-CNN, in: IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.

[7] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: IEEE Interna- tional Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.

[8] L. Jonathan, S. Evan, D. Trevor, Fully convolutional networks for seman- tic segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (4) (2017) 640–651.

[9] L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, DeepLab: Semantic image segmentation with deep convolutional nets, atrous con- volution, and fully connected CRFs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2018) 834–848.

[10] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the in- ception architecture for computer vision, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818–2826.

[11] S. Xie, R. Girshick, P. Dollar, Z. Tu, K. He, Aggregated residual transfor- mations for deep neural networks, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2017, pp. 5987–5995.

[12] L. Guosheng, M. Anton, S. Chunhua, I. Reid, RefineNet: multi -Path Re- finement Networks for High-Resolution Semantic Segmentation, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2017, pp. 5168–5177.

[13] B. Vijay, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder- decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (12) (2017) 2481–2495.

[14] H. Zhao, J. Shi, X. Qi, X. Wang, J.Y. Jia, Pyramid scene parsing network, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 6230–6239.

[15] J.D. Chen, Y.K. Cho, Z. Kira, Multi-view incremental segmentation of 3- D point clouds for mobile robots, IEEE Robot. Autom. Lett. 4 (2) (2019) 1240–1246.

[16] K.Q. Li, W.B. Tao, L.M. Liu, Online semantic object segmentation for vision robot collected video, IEEE Access 7 (2) (2019) 107602–107615.

[17] W. Chen, J. Wilson, S. Tyree, K. Weinberger, Y. Chen, Compressing neural networks with the hashing trick, in: International Conference on Machine Learning, 2015, pp. 2285–2294.

[18] S. Han, H. Mao, W.J. Dally, Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding, in: International Conference on Learning Representations, 2016, pp. 1–14.

[19] J. Wu, C. Leng, Y. Wang, Q. Hu, J. Cheng, Quantized convolutional neu- ral networks for mobile devices, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 5168–5177.

[20] M. Rastegari, V. Ordonez, J. Redmon, A. Farhadi, Xnor-net: Imagenet classification using binary convolutional neural networks, in: European Conference on Computer Vision, 2016, pp. 525–542.

[21] A. Paszke, A. Chaurasia, S. Kim, E. Culurciello, Enet: A deep neural network architecture for real-time semantic segmentation, 2016, arXiv preprint arXiv:1606.02147.

[22] E. Romera, J.M. Alvarez, L.M. Bergasa, R. Arroyo, ERFNet: Efficient residual factorized convnet for real-time semantic segmentation, IEEE Trans. Intell. Transp. Syst. 19 (1) (2018) 263–272.

[23] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: efficient convolutional neural networks for mobile vision applications, 2017, arXiv preprint arXiv:1704.04861.

[24] W. Wen, C. Wu, Y. Wang, Y. Chen, H. Li, Learning structured sparsity in deep neural networks, in: Annual Conference on Neural Information Processing Systems, 2016, pp. 2074–2082.

[25] B. Liu, M. Wang, H. Foroosh, M. Tappen, M. Pensky, Sparse convolutional neural networks, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2015, pp. 806–814.

[26] X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2018, pp. 6848–6856.

[27] M. Sandler, A. Howard, M.L. Zhu, A. Zhmoginov, L.C. Chen, Mobilenet V2: Inverted residuals and linear bottlenecks, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2019, pp. 4510–4520.

[28] N. Ma, X.Y. Zhang, H.T. Zheng, J. Sun, Shufflenet v2: Practical guidelines for efficient cnn architecture design, in: European Conference on Computer Vision, 2018, pp. 116–131.

[29] H.S. Zhao, X.J. Qi, X.Y. Shen, J.P. Shi, J.Y. Jia, Icnet for real -time semantic segmentation on high-resolution images, 2018, arXiv preprint arXiv:1704. 08545v2.

[30] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, H. Hajishirzi, ESPNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation, 2018, arXiv preprint arXiv:1803.06815v3.

[31] S. Mehta, M. Rastegari, L. Shapiro, H. Hajishirzi, ESPNet V2: A light-weight, power efficient, and general purpose convolutional neural network, 2019, arXiv:1811.11431v3.

[32] M.Y. Liu, H.J. Yin, Feature pyramid encoding network for real-time semantic segmentation, 2019, arXiv preprint arXiv:1909.08599v1.

[33] G. Li, I.Y. Yun, J.H. Kim, J.K. Kim, DABNet: depth-wise asymmetric bottleneck for real-time semantic segmentation, 2019, arXiv:1907.11357v1.

[34] S.Y. Lo, H.M. Hang, S.W. Chan, J.H. Lin, Efficient dense modules of asymmet- ric convolution for real-time semantic segmentation, 2018, arXiv preprint arXiv:1809.06323v2.

[35] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The scapes dataset for semantic urban scene understanding, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213–3223.

[36] G.J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla, Segmentation and recog- nition using structure from motion point clouds, in: European Conference on Computer Vision, 2008, pp. 44–57.

[37] S.R.B. G. Neuhold, P. Kontschieder, The mapillary vistas dataset for seman- tic understanding of street scenes, in:

IEEE International Conference on Computer Vision, 2017, pp. 4990–4999.
[38] R.P. Poudel, U.B.S. Liwicki, C. Zach, Contextnet: Exploring context and detail for semantic segmentation in real-time, 2018, arXiv preprint arXiv: 1805.04554.
[39] M.Y. Liu, H.J. Yin, Feature pyramid encoding network for real-time semantic segmentation, 2019, arXiv preprint arXiv:1909.08599v1.
[40] C.Q. Yu, J.B. Wang, C. Peng, C.X. Gao, G. Yu, N. Sang, Bisenet: Bilateral segmentation network for real-time semantic segmentation, in: European Conference on Computer Vision, 2018, pp. 325–341.
[41] T.Y. Wu, S. Tang, R. Zhang, Y.D. Zhang, CGNet: A light -weight context guided network for semantic segmentation, 2018, arXiv preprint arXiv: 1811.08201v1.
[42] G. Huang, S.C. Liu, L.V. der Maaten, K.Q. Weinberger, Condensenet: An efficient densenet using learned group convolutions, in: IEEE Interna- tional Conference on Computer Vision and Pattern Recognition, 2018, pp. 2752–2761.
[43] J.K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, Y. Bengio, Attention- based models for speech recognition, in: Annual Conference on Neural Information Processing Systems, 2015, pp. 577–585.
[44] K. Xu, B. Jimmy, K. Ryan, Show attend and tell: Neural image caption generation with visual attention, in: International Conference on Machine Learning, 2015, pp. 2048–2057.
[45] L.-C. Chen, Y. Yang, J. Wang, W. Xu, A.L. Yuille, Attention to scale: Scale- aware semantic image segmentation, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 3640–3649.
[46] F. Wang, M.Q. Jiang, C. Qian, S. Yang, C. Li, H.Q. Zhang, X.G. Wang, X.O. Tang, Residual attention network for image classification, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2017, pp. 6450–6458.
[47] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, 2017, arXiv preprint arXiv:1709.01507.
[48] H. Lu, D. Wang, Y. Li, J. Li, X. Li, H. Kim, S. Serikawa, I. Humar, CONet: A cognitive ocean network, IEEE Wirel. Commun. 26 (3) (2019) 90–96.
[49] X. Xu, H. Lu, J. Song, Y. Yang, H.T. Shen, X. Li, Ternary adversarial networks with self-supervision for zero-shot cross-modal retrieval, IEEE Trans. Cybern. 50 (6) (2020) 2400–2413.
[50] V. Mnih, N. Heess, A. Graves, Recurrent models of visual attention, in: Annual Conference on Neural Information Processing Systems, 2014, pp. 1–9.
[51] C.Q. Yu, J.B. Wang, C. Peng, C.X. Gao, G. Yu, N. Sang, Learning a discriminative feature network for semantic segmentation, in: IEEE Inter- national Conference on Computer Vision and Pattern Recognition, 2018, pp. 325–341.
[52] H. Zhang, K. Dana, J.P. Shi, Z.Y. Zhang, X.G. Wan, A. Tyagi, A. Agrawal, Context encoding for semantic segmentation, 2018, arXiv preprint arXiv: 1803.08904.
[53] Y. Wang, Q. Zhou, J. Liu, J. Xiong, G.W. Gao, X.F. Wu, L.J. Latecki, LEDNet: A lightweight encoder-decoder network for real-time semantic segmentation, in: IEEE International Conference on Image Processing, 2019, pp. 177–186.
[54] O. Ronneberger, F. Philipp, B. Thomas, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2015, pp. 225–233.
[55] H. Zhao, J. Shi, X. Qi, X. Wang, J.Y. Jia, Pyramid scene parsing network, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 6230–6239.
[56] C. Peng, Z. Xiangyu, Y. Gang, L. Guiming, S. Jian, Large kernel matters: Improve semantic segmentation by global convolutional network, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2017, pp. 1743–1751.
[57] H.C. Li, P.F. Xiong, J. An, L.X. Wang, Pyramid attention network for semantic segmentation, 2018, arXiv:1805.10180.
[58] Z.L. Zhang, X.Y. Zhang, C. Peng, X.Y. Xue, J. Sun, Exfuse: Enhancing feature fusion for semantic segmentation, in: European Conference on Computer Vision, 2018, pp. 269–284.
[59] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, in: International Conference on Learning Representations, 2016, pp. 1–10.
[60] A. Shrivastava, A. Gupta, R. Girshick, Training region-based object detectors with online hard example mining, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 761–769.
[61] R.P. Poudel, S. Liwicki, Fast-scnn: fast semantic segmentation network, 2019, arXiv:1902.04502v1.
[62] Y.H. Yuan, J.D. Wang, OCNet: Object context network for scene parsing, 2018, arXiv:1809.00916v1.
[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A large- scale hierarchical image database, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.
[64] L.Y. Liu, H.M. Jiang, P.C. He, W.Z. Chen, X.D. Liu, J.F. Gao, J.W. Han, On the variance of the adaptive learning rate and beyond, 2019, arXiv: 1908.03265v1.
[65] M. Zhang, J. Lucas, J. Ba, G.E. Hinton, Lookahead Optimizer: k steps forward, 1 step back, in: Annual Conference on Neural Information Processing Systems, 2019, pp. 9593–9604.