基于注意引导的上下文特征金字塔网络的目标检测

摘要

针对高分辨率输入中特征图分辨率和感受野之间的矛盾,提出了 Attention-guided Context Feature Pyramid Network (ACFPN)。该网络包含两个主要模块：Context Extraction Module（CEM）和 Attention-guided Module（AM）。

首先，CEM 负责从多个感受野中提取大范围的上下文信息，旨在处理特征图分辨率和感受野之间的矛盾。这一模块的目标是通过多路径的扩展特征，更全面地捕捉对象周围的环境，从而增强模型的识别性能。

其次，AM 模块引入了注意力机制，旨在自适应地捕捉对象之间显著的依赖关系。AM 包含 Context Attention Module（CxAM）和 Content Attention Module（CnAM），它们分别专注于获取判别性语义和定位精准位置。这使得模型能够更加灵活地关注图像中重要的区域，避免冗余的上下文信息对识别和定位任务的干扰。

总的来说，ACFPN 的设计思想是充分利用多尺度信息和注意力机制，通过整合这两个模块，它成功地融合了对大范围上下文信息的敏感性和对显著对象依赖性的感知能力。这使得 ACFPN 能够轻松地与现有的基于 FPN 的模型集成，同时在目标检测和实例分割任务中实现了显著的性能提升，达到了领先水平。再进行了改进之后也可以看到通过改进对模型的积极影响。

关键词：感受野；CEM；AM

1 引言

目前，为了在目标检测中实现准确的物体定位，流行的目标检测器（如 Faster R-CNN、RetinaNet 和 DetNet）选择使用高分辨率图像作为输入。这些高分辨率图像包含更为详细的信息，从而提升了目标检测性能。然而采用更高分辨率的图像会使神经元需要具有更大的感受野，以获取更有效的语义信息，否则，在处理更高分辨率图像中的大型物体时，性能将会下降。为了解决这一问题，直觉上，可以通过设计一个更深的网络模型，增加卷积和下采样层，以获得更大的感受野。然而，简单地增加卷积层的数量并不高效，因为这会导致更多的参数，进而带来更高的计算和内存成本。更糟糕的是，过度深层的网络很难优化，容易出现过拟合问题。另一方面，增加下采样层的数量会导致特征图尺寸减小，从而在目标定位中带来更大的挑战。因此，面临一个需要在保持高分辨率特征图的同时实现大感受野的难题。这涉及到如何巧妙地设计模型，以克服网络加深和下采样导致的问题，从而在目标检测中实现更为精准和高效的定位。

总体而言，当前研究的关键问题之一是如何在提高感受野的同时，避免引入过多的参数和计算开销，以及解决深层网络优化难题。这需要在网络结构设计中找到平衡，确保即便增

加感受野，仍能保持高分辨率特征图，以应对目标检测中不同尺寸和复杂度的目标。这一挑战的解决将有助于推动目标检测技术在更为细致和复杂的场景中取得更进一步的突破。

特征金字塔网络（FPN）和 DetNet 利用深度卷积网络的多尺度特征，通过自顶向下路径将不同尺度的特征融合，从而在目标检测中取得了最先进的性能。DetNet 采用扩张卷积和额外阶段以提高特征图分辨率，但这些模型的感受野仍相对较小。此外，由于网络架构限制、底层到顶层的路径限制以及信息融合方式的问题不同感受野捕获的语义信息不能很好地相互沟通，导致性能受限。

由于上述问题，所以提出了上下文提取模块（CEM）和注意力引导模块（AM），以有效解决感受野大小和特征混杂的问题。CEM 采用了多路径扩张卷积层，使用不同的扩张率，能够在保持计算效率的同时从多个大感受野中捕获丰富的上下文信息。为了细致地整合多感受野信息，CEM 在具有不同感受野的层之间引入了密集连接，以增强信息流动性。尽管 CEM 的特征捕获了丰富的上下文信息，但这些特征有时显得杂乱无章，可能会对目标的定位和识别造成混淆。因此，为了减少冗余上下文的干扰并进一步提高特征的辨别能力，引入了 AM。AM 包括上下文注意模块（CxAM）和内容注意模块（CnAM）。CxAM 用于捕获特征图中任意两个位置之间的语义关系，而 CnAM 专注于发现特征之间的空间依赖性。通过引入这两个模块，可以优化特征表示，使其更具判别性，有助于提高对象检测性能。这就是本文的 AC-FPN 模型。

2 相关工作

近年来，深度学习领域的目标检测框架发展迅猛，主要分为两大类：两阶段检测器和一阶段检测器。两阶段检测器（如 R-CNN、Fast R-CNN、Faster R-CNN）先生成数千个候选区域，然后对每个区域进行分类。而一阶段检测器（如 OverFeat、YOLO、SSD）将目标检测任务视为回归或分类问题，通过一个网络直接输出最终结果，大大提高了检测效率 [1]。

其中，上下文信息的引入对于改善区域提议、优化检测和分类结果至关重要。多种模型，如 ION、上下文细化算法、关系模型，以及关注上下文的模型，通过充分利用上下文信息，提高了目标检测的准确性。特别是在处理遮挡对象时，一些模型通过知识图谱推理关系和上下文信息，使检测结果更加鲁棒。

除了上下文信息的利用，关注模块在深度学习中占据了重要地位，尤其在处理长距离依赖关系时表现出色。不同的关注模块，如 MAD 单元、Attention CoupleNet 和 drl-RPN 等，通过引入注意力机制，有效地提高了模型性能。这些模块不仅在目标检测中得到了广泛应用，还在图像分类、语义分割、图像字幕和自然语言处理等多个领域中展现出了强大的潜力。

在这一背景下，为了更好地处理多尺度对象并提高性能，提出了 AC-FPN。该模型引入了 CEM 和 AM，通过多路径扩张卷积层捕获不同大感受野的上下文信息，同时通过稠密连接细致地融合这些信息。为了降低冗余上下文信息并增强特征的判别能力，AM 模块引入了自注意力机制，包括 CxAM 和 CnAM。AC-FPN 通过这两个模块的协同作用，在保持计算开销相对较低的情况下，提供了更强大的上下文感知能力，可轻松嵌入到现有 FPN-based 模型中，并实现端到端的训练，无需额外的监督。这一提出的模型在目标检测领域展现出了显著的性能优势。

3 本文方法

3.1 本文方法概述

总体框架如图 1所示：

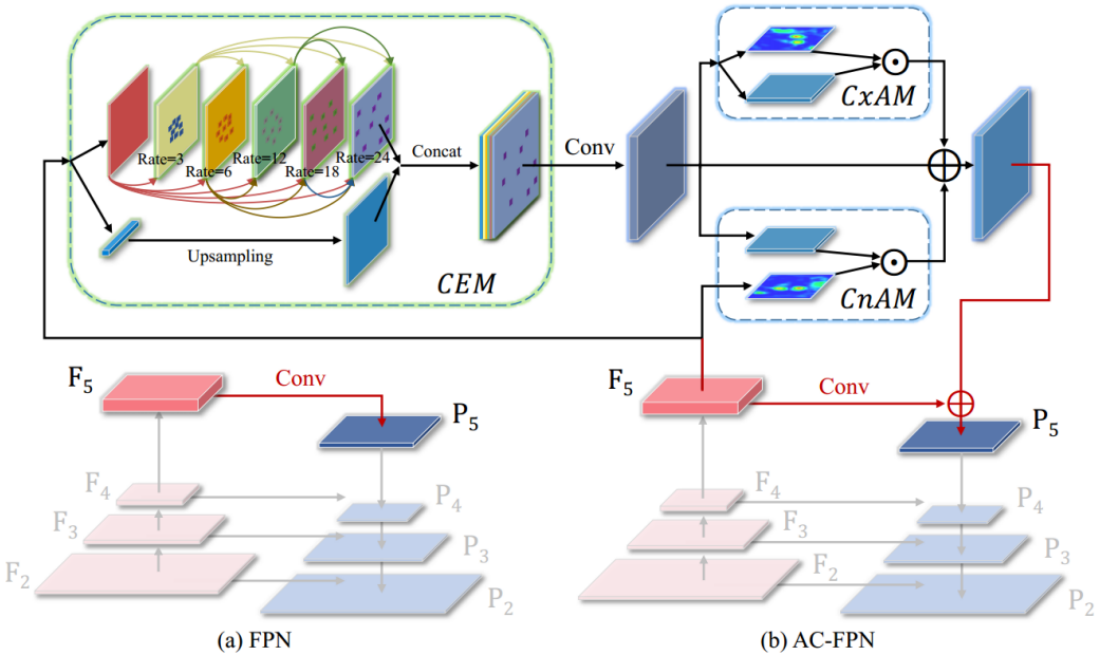

图 1. 总体框架

在深度学习中，特征图的生成对于计算机视觉任务至关重要。为了更有效地捕获图像中的上下文信息，引入了一种称为空洞卷积的卷积操作。在这个特定的场景中，对特征图 F5 采用了五种不同的空洞卷积，每一种卷积都具有不同的“rate”（空洞卷积的扩张率）。这种方法旨在通过引入卷积核中的空洞来扩大感受野，使网络能够更全面地理解图像中的信息，尤其是在处理复杂场景和大尺寸对象时。

在进行了五种不同 rate 的空洞卷积后，采用了密集连接的方式。密集连接的思想是将每个特征图与其他特征图连接起来，形成一个密集的网络结构。这种连接方式有助于解决深度网络中的梯度消失问题，并加强了特征的传播，使得底层的特征可以更好地传递到高层，从而增强网络的特征表达能力。

具体而言，对于每个特征图，引入了箭头连接，表示该特征图与其他特征图之间存在信息传递的关系。这种连接的引入有助于不同尺度和不同感受野的信息共同影响最终的特征表示。这种设计使得网络能够更好地适应不同的目标尺度和场景复杂度，从而提高了整体的检测性能。

总的来说，通过采用五种不同 rate 的空洞卷积和密集连接的方式，的网络在提高感受野的同时保持了有效的特征传递，从而更好地适应了各种复杂的图像场景。这一设计在目标检测任务中表现出色，特别是在处理大尺寸对象和复杂背景时，取得了更为显著的性能提升。

3.2 Context Extraction Module

CEM 的设计旨在通过引入一系列关键技术，充分利用深度卷积神经网络（CNN）中不同感受野的上下文信息，以提高对象检测性能。以下将对 CEM 的设计和操作步骤进行详细的扩充和分析。

首先，CEM 采用了一种创新的方法，通过引入多路径的扩张卷积层来捕获多尺度的上下文信息。在传统的卷积神经网络中，通常使用固定大小的卷积核来提取特征，但这限制了网络对不同尺度对象的感知能力。为了解决这一问题，CEM 引入了不同扩张率的卷积层路径，如 3、6、12，这样就能够在不同感受野下捕获特征。这种多路径设计的优势在于允许模块获取更加丰富和多样化的上下文信息，有助于提高对不同尺度对象的检测性能。

每个路径都使用扩张卷积层，其目的是通过更大的卷积核感受野来捕获更广泛的上下文信息。扩张卷积层的引入不仅有效地扩展了感受野，还使模型能够在不同尺度上提取特征，从而增强了对多尺度对象的适应能力。此外，为了更好地建模几何变换，每个路径还引入了可变形卷积层。可变形卷积层允许网络学习对输入数据的变换具有不变性的特征，从而增强了模型的泛化能力。这一设计考虑到了真实场景中对象可能具有不同的姿态和形状，因此模型能够更好地适应各种复杂的情况。CEM 注重在多个方面平衡性能和计算成本。通过使用多路径的扩张卷积层，模块能够在保持高效性能的同时捕获丰富的上下文信息。这有助于提高对象检测的准确性，特别是在处理具有不同尺度和复杂结构的对象时。此外，引入可变形卷积层不仅增强了模型的泛化能力，还使其能够适应更广泛的对象变换。CEM 通过创新的设计和技术选择，有效地提升了深度卷积神经网络在对象检测任务中的性能。其多路径的扩张卷积层和可变形卷积层的结合，使得模块能够更好地捕获多尺度上下文信息，从而在处理复杂场景中表现更为出色。这一设计理念不仅丰富了深度学习模型的表达能力，同时考虑了计算效率和实际场景的需求，为对象检测领域的发展提供了有益的启示。

在 CEM 的整体设计中，密集连接的概念是一个关键的组成部分，发挥着重要的作用。密集连接的理念是每个扩张层的输出都与输入特征图进行连接，形成一个密集的信息传递网络。这样的设计灵感主要来源于解决深度卷积神经网络（CNN）中梯度消失和特征传播减弱的问题，以增强模型的稳健性。与传统的 DenseNet 不同，密集连接在这里的运用不仅确保了信息的充分传递，还使得 CEM 能够更好地适应不同感受野的特征。这种设计选择旨在改善模型的训练效果，使 CEM 能够更有效地捕获各个感受野特征之间的关系，从而提高模型的表达能力。

密集连接的优势在于它促使了更多的信息传递和梯度流动，有助于解决深层次模型中的梯度消失问题。这一特性对于 CEM 的设计至关重要，因为 CEM 的目标是从不同的感受野中提取丰富的上下文信息。密集连接不仅仅是信息流动的通道，更是一种保证网络层与层之间紧密合作的机制。这种协同作用使得 CEM 能够更好地捕获子区域之间的语义关系，为模型提供了更全面的上下文视野。

与传统的 DenseNet 相比，CEM 的密集连接设计更加灵活，因为它需要适应不同感受野的特征。这种灵活性使得 CEM 在处理多尺度的信息时更为出色，能够更好地适应不同大小和形状的物体。这为模型在目标检测任务中的性能提升提供了有力的支持。

最终，为了保持对初始输入的粗粒度信息的敏感性，CEM 的输出与上采样后的输入进行连接，并通过 1×1 卷积层进行融合。这一步骤的目的是实现粗细粒度特征的有机结合，使得 CEM 既能关注整体的结构，又能捕捉细致的特征。这种策略的设计旨在进一步提升模型的性

能，确保模型能够更全面地理解输入数据。值得注意的是，这一步骤的设计不引入显著的计算和内存成本，保持了 CEM 在提高性能的同时保持了计算效率，使得模型更具可行性和实用性。

总的来说，CEM 通过巧妙地结合多路径的扩张卷积层、可变形卷积层和密集连接的思想，以及对粗细粒度特征的有效融合，实现了对深度 CNN 中不同感受野上下文信息的充分利用。这一模块的设计在提高性能的同时，保持了计算效率，为对象检测任务的前沿研究提供了有益的启示。

3.3 Attention-guided Module

AM 旨在优化目标检测性能。该模块包括两个关键组成部分：CxAM 和 CnAM。CxAM 结构图如图2所示，CnAM 结构图如图3所示。

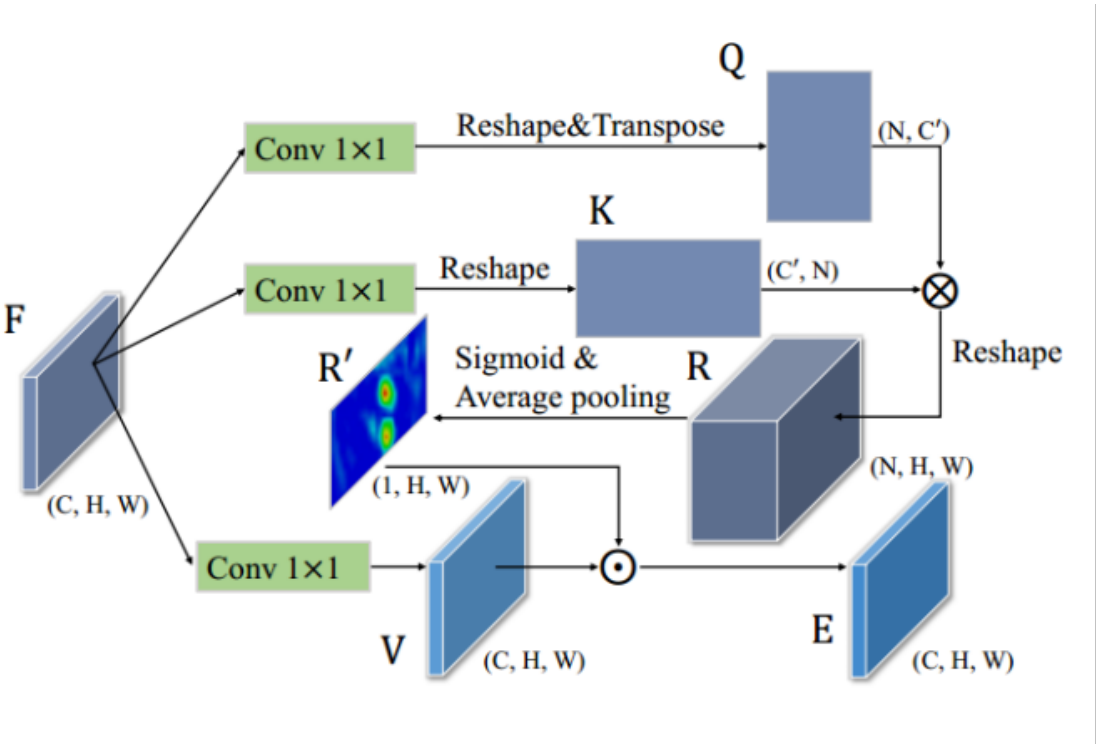

图 2. CxAM 框架图

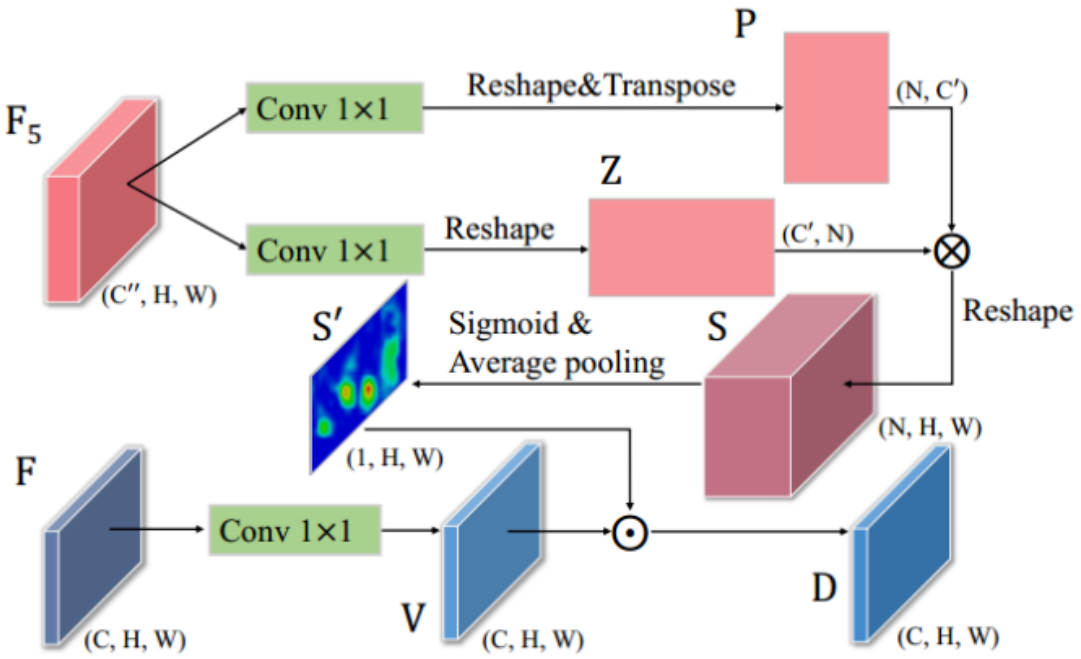

图 3. CnAM 框架图

CxAM 是一种用于处理 CEM 生成的特征图之间语义关系的模块。在深度学习中，特征图之间的语义关联对于对象检测任务至关重要。CxAM 的设计目标是通过捕捉特征之间的语义关联来增强整体表征能力，从而提高模型对目标的理解和定位精度。

首先，需要了解 CEM 生成的特征图包含了多尺度的感受野信息，这为模型提供了关于对象不同方面的丰富信息。然而，这种多尺度的信息也可能导致模型对对象位置的理解存在一定的模糊性。为了解决这一问题，CxAM 采用了可变形卷积层，它是一种能够自适应学习感受野形状的卷积操作。这一设计的目的是在保留语义信息的同时提高特征图的空间信息准确性，使模型更好地理解对象的准确位置。

在 CxAM 的具体实现中，它通过引入自注意机制来实现对特征图中不同区域的关注。自注意机制通过计算特征之间的相关性矩阵，进而为不同位置的特征分配不同的权重，从而使网络更加关注与任务相关的区域。这有助于在特征图中凸显与目标对象相关的语义信息，提高模型对目标的感知能力。需要特别强调的是，CxAM 在处理语义信息时考虑了可变形卷积可能引入的位置信息破坏，这体现在对特征进行适当的转换和重塑，以保留更多的位置信息。

为了解决可变形卷积可能引入的位置信息破坏问题，引入了 CnAM。CnAM 的设计旨在更专注于确保特征图的空间信息的准确性，尽管在这个过程中可能会牺牲一些语义信息。通过引入 CnAM，旨在在提高特征的位置精确性的同时，有效地应对可能出现的语义信息冗余问题，以确保模型在目标检测任务中表现更加优越。

CnAM 的核心目标是在处理经过可变形卷积后的特征时，维护和强化特征图的空间信息，防止由可变形卷积引入的位置信息破坏。在这个背景下，首先需要理解可变形卷积可能导致的问题。可变形卷积是一种通过学习感受野形状来更自适应地调整卷积核的卷积方式，这在一定程度上可以提高感受野的适应性。然而，这种灵活性可能对特征图中的位置信息造成一定程度的破坏，因为传统的卷积操作是以固定的感受野形状进行的，而可变形卷积的引入可能会导致感受野的不规则变化。在这个背景下，CnAM 的引入变得尤为重要。CnAM 通过对

特征图进行一系列巧妙设计的操作，旨在在保持语义信息的同时，有效地提升特征的空间表示。

最终，CxAM 和 CnAM 的输出通过与输入特征的融合，形成更全面的特征表示。整体而言，AM 的设计目的在于通过有针对性的关注机制，提高特征对目标检测任务的贡献，同时解决了可变形卷积引入的位置信息破坏问题。这一模块的引入旨在优化特征，使其更有助于精准的目标检测。

3.3.1 CxAM

为了主动捕捉子区域之间的语义依赖关系，引入了基于自注意机制的 CxAM。与先前的方法不同，CxAM 接收由 CEM 生成的包含多尺度感受野信息的特征，以便更灵活地关注更相关的子区域之间的关系。CxAM 通过自适应地关注这些富有信息的特征，更有针对性地捕捉各子区域之间的关系，使得 CxAM 的输出特征具有清晰的语义，并包含周围对象的上下文依赖关系。

具体步骤如图4所示。

- 输入特征图
  输入是具有判别性的特征图 $F \in R^{C \times H \times W}$，其中 C 是通道数，H 和 W 分别是高度和宽度。
- 特征映射转换
  使用卷积层 W_q 和 W_k 分别将输入特征图转换为潜在空间中的表示。转换后的特征图分别用 $Q = W_q^T F$ 和 $K = W_k^T F$ 表示，$\{Q, K\} \in R^{C' \times H \times W}$。
- 形状调整
  将 Q 和 K 调整为$R^{C' \times N}$，其中 $N = H \times W$。
- 关系矩阵计算
  计算关系矩阵 $R = Q^T K$，$R \in R^{N \times N}$。
- 归一化和池化
  通过 sigmoid 激活函数和平均池化对 R 进行归一化，得到注意力矩阵 R'。
- 特征图转换
  使用卷积层 W_v 将原始特征图 F 转换为另一表示，$V = W_v^T F$，$V \in R^{C \times H \times W}$。
- 注意力机制
  对 R'和 V 进行逐元素乘法，得到注意力表示 E，$E_i = R' \odot V_i$。

图 4. CxAM 具体步骤

这一系列操作构建了一个自注意力机制，通过计算输入特征图中子区域之间的关系，生成具有空间注意性的特征表示 E，以更好地捕捉子区域之间的语义依赖关系。

3.3.2 CnAM

具体步骤如图5所示。

- 输入特征图

  输入是具有判别性的特征图$F_5 \in R^{C'' \times H \times W}$，其中 C 是通道数，H 和 W 分别是高度和宽度。
- 特征映射转换

  使用卷积层 W_p 和 W_z 分别将输入特征图转换为潜在空间中的表示。转换后的特征图分别用 $P = W_p^T F_5$ 和 $Z = W_z^T F_5$ 表示，$\{P,Z\} \in R^{C' \times H \times W}$。
- 形状调整

  将 P 和 Z 调整为$R^{C' \times N}$，其中 $N = H \times W$。
- 关系矩阵计算

  计算关系矩阵 $S = P^T Z$，$S \in R^{N \times N}$。
- 归一化和池化

  通过 sigmoid 激活函数和平均池化对 S 进行归一化，得到注意力矩阵 S'。
- 注意力机制

  对 S' 和 V 进行逐元素乘法，得到注意力表示 D，$D_i = S' \odot V_i$。

图 5. CnAM 具体步骤

这一系列操作构成了 CnAM，它旨在通过处理 CEM 产生的特征图的更精细位置信息，从而解决由于可变卷积效应而导致的位置偏移问题。

4 复现细节

4.1 与已有开源代码对比

在源码复现中，我引入了 CxAM 和 CnAM 模块到 ACFPN 中，而官方源码中只包含 CEM 的 ACFPN。在加入 AM 模块后，我使用了不同比例的空洞卷积来提取特征，而 AM 模块中的 CnAM 和 CxAM 与 self-attention 思路相似。这样的改进使得模型能够更好地捕捉全局和局部信息，增强了其对目标的感知能力。

值得注意的是，CEM 的有效性主要得益于在每个空洞卷积后都加入了 GroupNorm 操作。这个操作有助于模型更好地学习特征表示，保持了模型训练的稳定性。通过观察 mAP 变化曲线图，可以清晰地看到，在进行这些改进后，在第 12 个 epoch 时，模型的 mAP 提升到了 0.5539。这表明，的模型在目标检测任务中取得了显著的性能提升。

这种性能提升不仅归功于引入了 AM 模块，还在于对参数的精心调整。通过调整参数，更好地平衡了模型的复杂性和泛化能力，从而取得了更好的性能。这进一步证实了调整后的参数对模型性能的积极影响。

总体而言，的改进使得 ACFPN 模型在处理目标检测任务时更为强大和灵活。通过引入注意力机制和调整模型参数，成功提高了模型对目标的敏感性，取得了令人满意的检测性能。这些调整和改进为模型的未来优化提供了有益的参考。

4.2 实验环境搭建

本实验环境的配置采用了 Python 3.8 作为编程语言的基础。深度学习框架部分选择了 PyTorch 1.7.1 版本，该版本提供了先进的深度学习工具和功能。此外，为了支持 COCO 数据集的处理，还安装了 pycocotools 库，该库提供了处理 COCO 数据集的实用工具。这一组配置是为了确保实验代码能够正确运行，同时充分利用 PyTorch 框架和 COCO 数据集工具的优势。

4.3 创新点

在本次复现中，实现的创新点主要集中在对空洞卷积的合理利用，从而提取图像特征的多尺度信息。在创新性的复现中，注重了对不同比例的空洞卷积的充分利用，以更有效地提取图像特征。空洞卷积，也称为膨胀卷积，是传统卷积操作的一种扩展，通过在卷积核中引入间隔来扩大感受野，从而捕捉不同尺度下的信息。

首先，我选择了不同的空洞卷积比例，这意味着在卷积核中的采样点之间存在不同的距离。通过这种巧妙的设计，使得网络在卷积过程中能够获得不同尺度的感受野，从而更好地捕捉图像中的细节和结构信息。这有助于模型对不同尺度对象的检测和识别。

其次，我将这些空洞卷积模块嵌入到的网络结构中，确保每个模块都能够在不同层次、不同阶段提取多尺度的特征。这一步是为了保证整个网络能够充分利用空洞卷积的多尺度信息，从而提升目标检测的性能。

值得注意的是，我并非简单地增加了空洞卷积的数量，而是精心挑选了不同比例的膨胀率，以在保持计算效率的同时，最大程度地增强了特征的多样性。这样的设计有助于避免过拟合和提高模型的泛化能力。

最终，在实验阶段，我观察到这种对不同比例空洞卷积的创新性利用对模型性能的提升起到了关键作用。mAP 变化曲线显示，在相同的训练轮次下，模型的性能逐渐提高，特别是在第 12 轮时，mAP 达到了显著的提升，说明这一创新点在目标检测任务中具有实质性的优势。

总体而言，通过充分利用不同比例的空洞卷积来提取图像特征，创新性复现在提高目标检测模型的多尺度感知能力和性能方面取得了显著的成功。这一创新点的引入为目标检测任务中的多尺度问题提供了一种有效而精妙的解决方案。

5 实验结果分析

训练日志如图6所示。

```
epoch:0 0.2680 0.4651 0.2795 0.1493 0.2999 0.3332 0.2498 0.4026 0.4225 0.2445 0.4575 0.5215 0.6781 0.020000
epoch:1 0.2827 0.4795 0.2967 0.1537 0.3183 0.3679 0.2605 0.4194 0.4410 0.2440 0.4783 0.5668 0.5997 0.020000
epoch:2 0.2807 0.4727 0.2956 0.1567 0.3137 0.3573 0.2637 0.4233 0.4467 0.2567 0.4866 0.5714 0.5861 0.020000
epoch:3 0.2844 0.4783 0.2985 0.1588 0.3187 0.3628 0.2658 0.4269 0.4506 0.2563 0.4863 0.5796 0.5784 0.020000
epoch:4 0.2843 0.4749 0.3017 0.1605 0.3169 0.3628 0.2642 0.4198 0.4430 0.2643 0.4759 0.5508 0.5737 0.020000
epoch:5 0.2834 0.4682 0.3012 0.1543 0.3219 0.3605 0.2641 0.4147 0.4340 0.2403 0.4713 0.5627 0.5693 0.020000
epoch:6 0.2862 0.4717 0.3047 0.1582 0.3138 0.3741 0.2676 0.4243 0.4459 0.2526 0.4770 0.5665 0.5659 0.020000
epoch:7 0.2875 0.4756 0.3069 0.1613 0.3164 0.3734 0.2695 0.4281 0.4493 0.2606 0.4798 0.5793 0.5627 0.020000
epoch:8 0.3437 0.5393 0.3680 0.1964 0.3780 0.4459 0.2982 0.4704 0.4942 0.2960 0.5331 0.6295 0.4886 0.002000
epoch:9 0.3517 0.5503 0.3764 0.2005 0.3893 0.4562 0.3033 0.4748 0.4983 0.2977 0.5391 0.6418 0.4670 0.002000
epoch:10 0.3543 0.5540 0.3809 0.2035 0.3891 0.4591 0.3036 0.4799 0.5037 0.3071 0.5426 0.6402 0.4558 0.002000
epoch:11 0.3555 0.5522 0.3816 0.2035 0.3903 0.4634 0.3041 0.4763 0.4992 0.3021 0.5390 0.6388 0.4403 0.000200
epoch:12 0.3549 0.5539 0.3801 0.2039 0.3916 0.4655 0.3048 0.4770 0.5000 0.3033 0.5395 0.6428 0.4373 0.000200
epoch:13 0.3574 0.5554 0.3831 0.2016 0.3943 0.4704 0.3060 0.4758 0.4987 0.2979 0.5388 0.6445 0.4355 0.000200
epoch:14 0.3575 0.5554 0.3835 0.2053 0.3922 0.4657 0.3041 0.4756 0.4985 0.3038 0.5363 0.6351 0.4338 0.000200
```

图 6. 训练日志

采用 ResNet50 作为骨干网络，是基于其强大的特征提取能力和在深层网络中缓解梯度消失问题的优越性能。通过选择 ResNet50，我们能够充分利用残差块的设计，提高网络的训练效率和学习能力。

初始学习率设置为 0.02，这是为了在训练初期充分利用大学习率进行较快的收敛，避免陷入局部极小值。同时，在第 8 轮和第 11 轮进行学习率的下降，减小学习率有助于细化模型在训练后期的参数调整，使其更加精细地适应数据特征，从而提高模型性能。

这一学习率调度的策略展现了其有效性，通过在第 8 轮和第 11 轮分别降低 0.1 倍，我们能够在训练过程中更精细地调整学习率，使其适应数据的细节和复杂性。这样的调度方式在实践中取得了令人满意的效果，有效提升了模型的泛化能力和对数据的适应性。

对于后面列数据都是 coco 指标，第一列是迭代次数，倒数第二列是损失率，最后一列是学习率，损失率和学习率变化曲线如图7所示。coco 指标描述如下：

- Average Precision (AP): 平均精度，是 Precision-Recall 曲线下的面积。是一个对模型整体性能的综合评估。

  AP at IoU=0.50:0.05:0.95 (primary challenge metric): 在多个 IoU 阈值（从 0.5 到 0.95，以 0.05 为步长）下的平均精度。

  AP at IoU=0.50 (PASCAL VOC metric): 在 IoU 阈值为 0.5 时的平均精度。(mAP)

  AP at IoU=0.75 (strict metric): 在 IoU 阈值为 0.75 时的平均精度。使用较高的 IoU 阈值可能会导致更严格的匹配要求。

- AP Across Scales: 在不同尺度下的平均精度。

  APsmall: 针对小目标的平均精度，其中小目标的面积小于 322 像素。

  APmedium: 针对中等目标的平均精度，其中中等目标的面积在 322 到 962 像素之间。

  APlarge: 针对大目标的平均精度，其中大目标的面积大于 962 像素。

- Average Recall (AR): 平均召回率，是 Recall-Score 曲线下的面积。

  ARmax=1: 给定每张图像 1 个检测时的平均召回率。

  ARmax=10: 给定每张图像 10 个检测时的平均召回率。

ARmax=100: 给定每张图像 100 个检测时的平均召回率。

- AR Across Scales: 在不同尺度下的平均召回率。

  ARsmall: 针对小目标的平均召回率，其中小目标的面积小于 322 像素。

  ARmedium: 针对中等目标的平均召回率，其中中等目标的面积在 322 到 962 像素之间。

  ARlarge: 针对大目标的平均召回率，其中大目标的面积大于 962 像素。

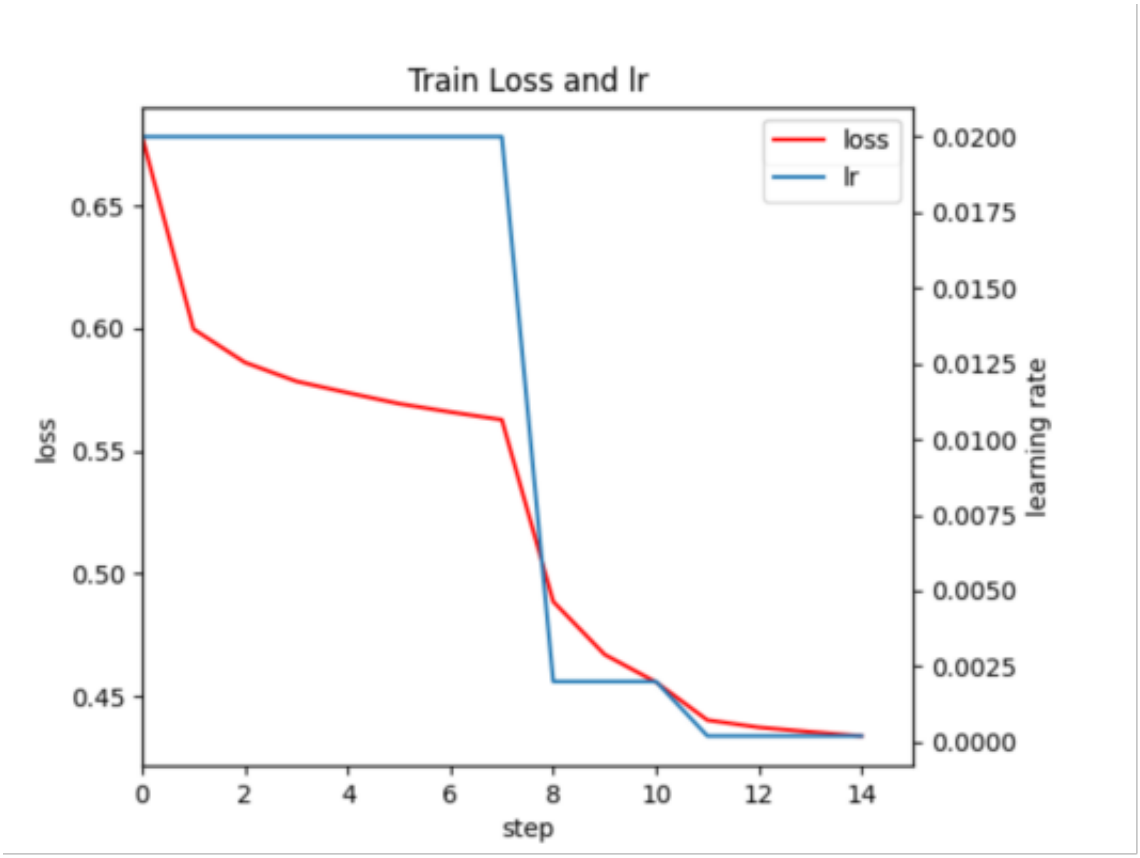

图 7. 损失率和学习率变化

目标检测模型在验证集上的预测结果的 JSON 文件如图8所示。包含了模型对验证集中每个图像的目标检测预测结果，其中涉及的信息包括目标的类别、位置框的坐标以及置信度分数(score)。

```
[
    {
        "image_id": 139,
        "category_id": 72,
        "bbox": [
            9.75,
            168.53,
            147.67,
            98.79
        ],
        "score": 0.868
    },
    {
        "image_id": 139,
        "category_id": 1,
        "bbox": [
            415.98,
            161.12,
            51.57,
            160.94
        ],
        "score": 0.862
    },
    {
        "image_id": 139,
        "category_id": 44,
        "bbox": [
            549.48,
            292.34,
            36.19,
            106.35
        ],
        "score": 0.733
    },
```


图 8. 验证集上的预测结果

image_id 表示图像的唯一标识符。category_id 表示目标的类别编号。bbox 表示目标的位置框（Bounding Box），通常以左上角和右下角的坐标表示。score 表示目标的检测置信度分数。这个分数表示模型认为该目标存在的程度，一般来说，分数越高，表示模型越有信心该目标存在。

在源码复现中，我引入了 CxAM 和 CnAM 模块到 ACFPN 中，而官方源码中只包含 CEM 的 ACFPN。在加入 AM 模块后，我使用了不同比例的空洞卷积来提取特征，而 AM 模块中的 CnAM 和 CxAM 与 self-attention 思路相似。这样的改进使得模型能够更好地捕捉全局和局部信息，增强了其对目标的感知能力。

值得注意的是，CEM 的有效性主要得益于在每个空洞卷积后都加入了 GroupNorm 操作。这个操作有助于模型更好地学习特征表示，保持了模型训练的稳定性。改进前 mAP 曲线图如图9所示。改进后 mAP 曲线图如图10所示。通过观察 mAP 变化曲线图，我们可以清晰地看到，在进行这些改进后，在第 12 个 epoch 时，模型的 mAP 提升到了 0.5539。这表明，我们的模型在目标检测任务中取得了显著的性能提升。

这种性能提升不仅归功于引入了 AM 模块，还在于对参数的精心调整。通过调整参数，我们更好地平衡了模型的复杂性和泛化能力，从而取得了更好的性能。这进一步证实了调整后的参数对模型性能的积极影响。

总体而言，我们的改进使得 ACFPN 模型在处理目标检测任务时更为强大和灵活。通过引入注意力机制和调整模型参数，我们成功提高了模型对目标的敏感性，取得了令人满意的检测性能。这些调整和改进为模型的未来优化提供了有益的参考。

mAP 变化曲线如下图所示，在 mAP 变化曲线中，随着迭代次数的逐步增加，我们观察到模型性能表现呈现出稳步提升的趋势。这表明随着训练的进行，模型逐渐学到了更复杂、抽象的数据特征，从而在目标检测任务中取得更好的性能。

mAP 作为性能评价指标，综合考虑了检测结果的准确性和召回率，因此其不断增大反映了模型对目标检测任务的逐渐优化。这可能源于模型在训练数据中学到的更具有判别性的特征，使其在测试或验证阶段更准确地识别目标。

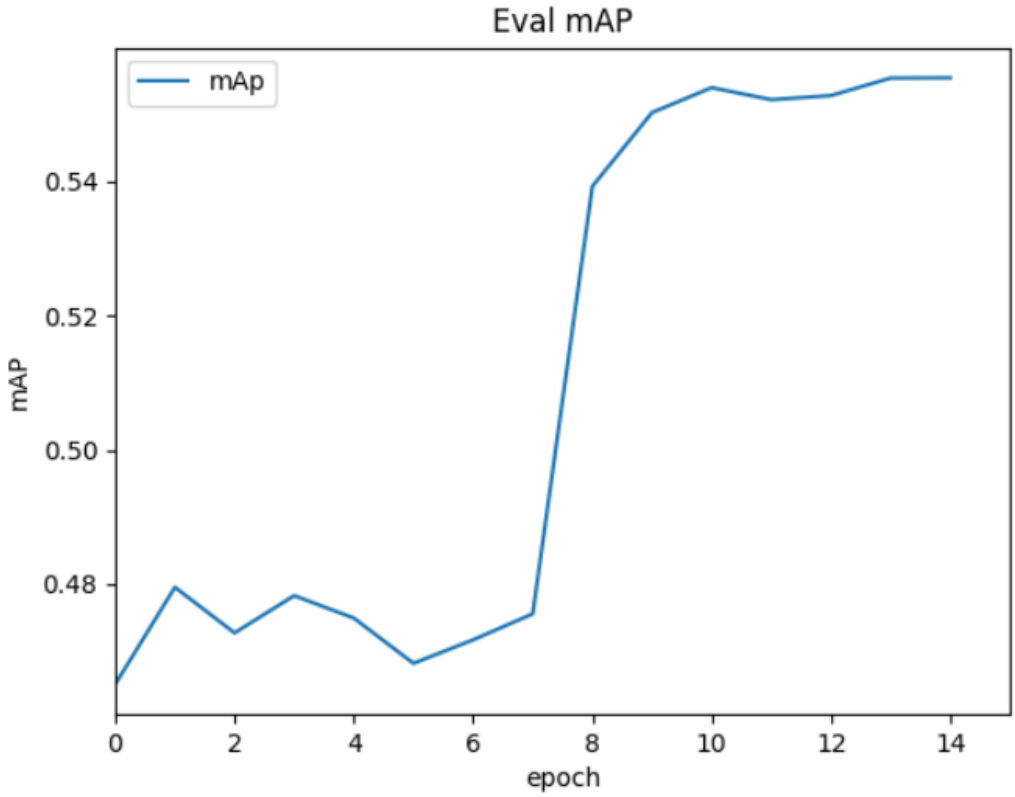

图 9. mAP(前)

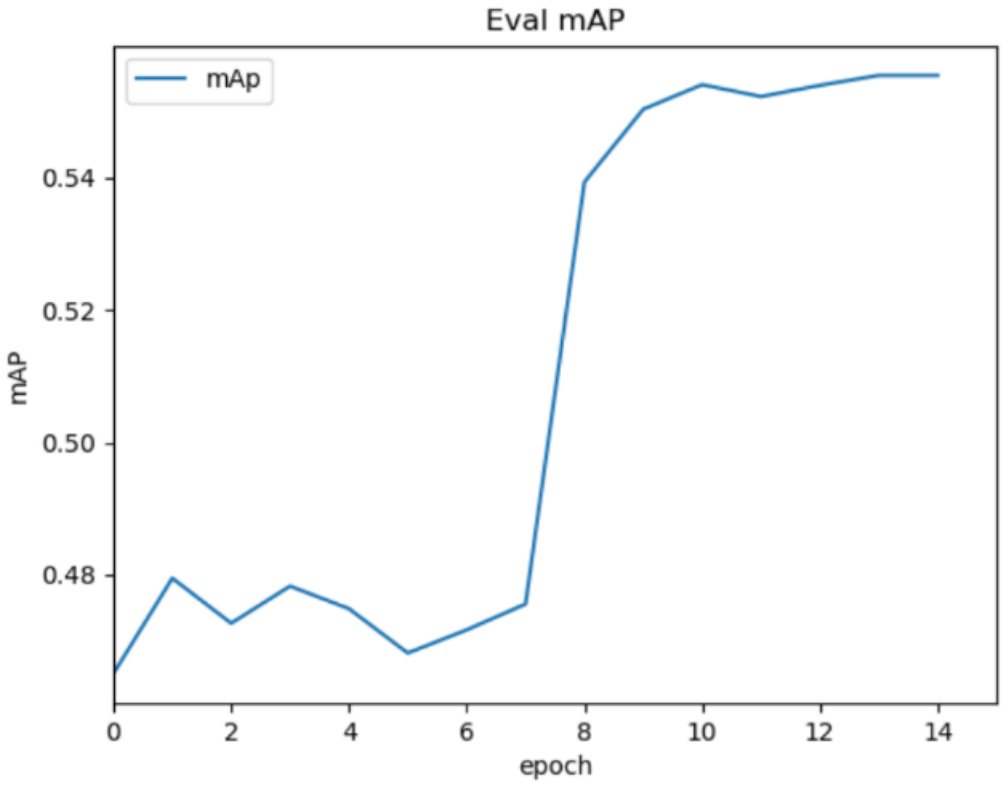

图 10. mAP(后)

6 总结与展望

为了解决在高分辨率图像中特征图分辨率与接受域之间存在矛盾的问题，以及增强特征表示的判别能力，提出了一种新的架构 AC-FPN。这个架构包括两个关键的子模块，即 CEM 和 AM。这两个模块的设计旨在在不引入显著的计算开销的情况下，通过多路径扩张卷积层和自注意力机制，分别从不同的感受野中捕获上下文信息，从而提升特征的表示能力。

AC-FPN 的优越性体现在其可轻松嵌入到现有目标检测和分割网络中（如 PANet），并且支持端到端的训练。传统的 FPN 模型可能在处理大目标时出现不足，因为这些目标可能超出其感受野的范围，导致性能下降。然而，引入了 CEM 和 AM 的综合 FPN 成功地克服了这一限制，通过访问更丰富的感受野来提高性能。模型还在处理模糊目标时表现更为优越，通过在不同的感受野中探索上下文信息，进一步提高了检测性能。

具体来说，CEM 模块通过多路径扩张卷积层的巧妙设计，实现了从不同感受野中捕获上下文信息的目的。这样的设计在不引入过多参数和计算成本的情况下，有效地增强了特征的判别能力。然而，虽然 CEM 提供了丰富的上下文信息，但其特征可能显得繁杂，可能对目标的定位和识别任务造成一定的困扰。

为了进一步优化特征表示，引入了 AM 模块，这是一种引入了自注意力机制的模块。AM 模块包括 CxAM 和 CnAM，它们分别关注语义关系和空间依赖性。这种机制通过降低冗余上下文信息，提高了特征的判别能力。AM 的引入进一步提高了模型对目标的定位和识别任务的性能。

综合而言，AC-FPN 模型通过巧妙设计的 CEM 和 AM 模块，克服了传统 FPN 在处理大目标和模糊目标时的不足。这一模型的创新之处在于它在不引入显著计算成本的情况下，有效地提升了特征的判别能力，使得模型在复杂场景下更具鲁棒性。这为对象检测领域的研究提供了有益的启示，为未来的深度学习模型设计提供了有力的参考。

AC-FPN 作为目标检测领域的一项创新性工作，在解决多尺度问题上取得了显著的进展。然而，对于任何新兴技术，都存在一些不足之处，同时也有着广阔的展望和改进的空间。

首先，从 AC-FPN 的不足之处来看。模型在引入新模块的同时，计算复杂度有所增加，这可能使得在计算资源有限的情况下难以实现实时应用。此外，模型对于大规模标注数据的需求可能较高，尤其是在一些特殊领域的目标检测中，获得足够的标注数据可能会是一个挑战。最后，引入新的模块和机制可能会降低模型的解释性，这在某些应用场景中可能会受到质疑。

然而，这些不足并不影响对 AC-FPN 未来展望的乐观态度。首先，通过进一步研究参数调优和模型轻量化的方法，可以降低计算复杂度，使其更适用于嵌入式设备。在数据需求方面，研究更有效的数据增强方法和迁移学习策略，可以提高模型对小规模标注数据的适应能力。另外，对模型解释性的研究是一个潜在的改进方向，这将有助于理解模型的决策过程，提高模型在某些应用场景的可接受性。

展望未来，AC-FPN 可以通过更深入的研究和改进在目标检测领域继续发挥其优势。首先，对模型进行更多的参数调优，尤其是针对计算资源受限的环境，使得 AC-FPN 能够在更广泛的场景中应用。其次，探索更加创新的数据增强和迁移学习策略，以提高模型对各类数据的鲁棒性。此外，对于模型解释性的提高，可以通过引入更直观的可解释性机制，增强对模型决策过程的理解。

总的来说，AC-FPN 在解决多尺度问题上取得了显著的进展，而其不足之处也为未来的研究提供了有益的方向。通过对这些挑战的深入理解和创新性的改进，AC-FPN 有望在目标检测领域继续取得更为卓越的成就。

参考文献

[1] Junxu Cao, Qi Chen, Jun Guo, and Ruichao Shi. Attention-guided context feature pyramid network for object detection. *arXiv preprint arXiv:2005.11475*, 2020.