

RT-DETRv3: 基于层次化密集正监督的实时端到端目标检测

王硕* 夏春龙* 吕峰 史一峰† 百度公司, 中国

{哇 ngshuo36、xiachunlong、lvfeng02、shiyifeng}@bai

都网

摘要

RT-DETR is the first real-time end-to-end transformer-based object detector. Its efficiency comes from the framework design and the Hungarian matching. However, compared to dense supervision detectors like the YOLO series, the Hungarian matching provides much sparser supervision, leading to insufficient model training and difficult to achieve optimal results. To address these issues, we proposed a hierarchical dense positive supervision method based on RT-DETR, named RT-DETRv3. Firstly, we introduce a CNN-based auxiliary branch that provides dense supervision that collaborates with the original decoder to enhance the encoder's feature representation. Secondly, to address insufficient decoder training, we propose a novel learning strategy involving self-attention perturbation. This strategy diversifies label assignment for positive samples across multiple query groups, thereby enriching positive supervisions. Additionally, we introduce a shared-weight decoder branch for dense positive supervision to ensure more high-quality queries matching each ground truth. Notably, all aforementioned modules are training-only. We conduct extensive experiments to demonstrate the effectiveness of our approach on COCO val2017. RT-DETRv3 significantly outperforms existing real-time detectors, including the RT-DETR series and the YOLO series. For example, RT-DETRv3-R18 achieves 48.1% AP (+1.6%/+1.4%) compared to RT-DETR-R18/RT-DETRv2-R18, while maintaining the same latency. Furthermore, RT-DETRv3-R101 can attain an impressive 54.6% AP outperforming YOLOv10-X. The code will be released at <https://github.com/clxia12/RT-DETRv3>.

1. 引言

目标检测是计算机视觉中的一个重要基础问题, 主要致力于获取{v*}

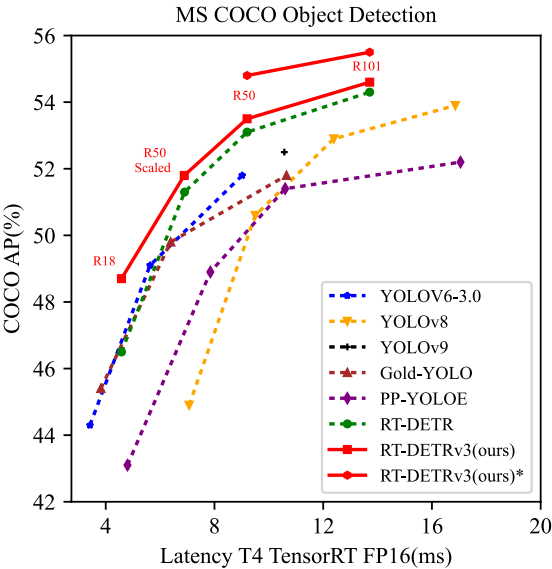


图1. 与其他实时目标检测器相比, 我们的方法在速度与精度权衡上表现更优。*表示额外添加数据的情况。

图像中物体的位置与类别信息。实时目标检测对算法性能有着更高要求, 例如推理速度需超过30帧/秒。其在自动驾驶、视频监控、目标跟踪等实际应用中具有巨大价值。近年来, 实时目标检测凭借高效的推理速度和卓越的检测精度, 引起了研究人员与行业专业人士的高度关注。其中最为流行的是基于CNN的单阶段实时检测器, 如YOLO系列([1, 11–13, 22, 24, 26])。它们均采用一对多的标签分配策略, 设计了高效推理框架, 并利用非极大值抑制(NMS)过滤冗余预测结果。尽管该策略会引入额外延迟, 但实现了精度与速度的平衡。

*Equal Contribution.

†Corresponding Author.

DETR [2]是首个基于Transformer的端到端目标检测算法。它采用集合预测方式，并通过匈牙利匹配策略进行优化，无需NMS后处理步骤，从而简化了目标检测流程。后续的DETR变体（如DAB-DETR [16]、DINO [29]和IDN-DETR [14]等）进一步引入了迭代优化方案和去噪训练机制，有效提升了模型收敛速度与检测性能。然而，其高昂的计算复杂度极大限制了实际应用场景。

RT-DETR[32]是首个基于Transformer的实时端到端目标检测算法。它设计了高效的混合编码器、IoU感知查询选择模块和可扩展的解码器层，取得了优于其他实时检测器的效果。然而，匈牙利匹配策略在训练过程中提供的监督信号较为稀疏，导致编码器和解码器训练不足，限制了该方法的性能上限。RT-DETRv2[19]通过优化训练策略，在不牺牲速度的前提下提升性能，进一步增强了RT-DETR[32]的灵活性和实用性，但需要更长的训练时间。为解决目标检测中监督稀疏性问题，我们提出了一种层次化密集正监督方法，通过在训练时引入多个辅助分支，有效加速模型收敛并提升性能。主要贡献如下：

- 我们引入了一个基于CNN的一对多标签分配辅助头，它与原始检测分支协同优化，进一步增强了编码器的表征能力。
- 我们提出了一种带有自注意力扰动的学习策略，旨在通过多样化多个查询组间的标签分配来增强解码器的监督效果。此外，我们引入了一个共享权重的解码器分支，用于密集正样本监督，以确保更多高质量查询与每个真实标注匹配。这些方法显著提升了模型性能，并在不增加推理延迟的情况下加速了收敛过程。
- 在COCO数据集上进行的大量实验充分验证了我们所提方法的有效性。如图1所示，RT-DETRv3显著优于其他实时检测器，包括RT-DETR系列和YOLO系列。例如，RT-DETRv3-R18在保持相同延迟的同时，AP达到48.1% (+1.6%)，优于RT-DETR-R18。此外，RT-DETRv3-R50的AP比YOLOv9-C高出0.9%，同时延迟还降低了1.3毫秒。

2. 相关工作

2.1. 基于CNN的实时目标检测。

当前基于CNN的实时目标检测器主要为YOLO系列。YOLOv4[1]和YOLOv5[12]优化了网络架构（如采用CSPNet[25]和PAN[17]），同时运用了Mosaic数据增强技术。YOLOv6[13]进一步优化了结构，包括RepVGG[6]骨干网络、解耦头、SimSPPF模块，以及更高效的训练策略（如SimOTA[7]等）。YOLOv7[24]引入E-ELAN注意力模块以更好地融合多层次特征，并采用自适应锚框机制提升小目标检测性能。YOLOv8[11]提出C2f模块实现高效特征提取与融合。YOLOv9[26]创新性地提出GELAN架构，并设计PGI机制以增强训练过程。PP-YOLO系列[10,18]是百度基于PaddlePaddle框架提出的实时目标检测方案，该系列算法在YOLO基础上进行优化改进，旨在提升检测精度与速度，满足实际应用场景需求。

2.2. 基于Transformer的实时目标检测

RT-DETR[32]是首个实时端到端目标检测器。该方法设计了一种高效的混合编码器，通过解耦尺度内交互与跨尺度融合，有效处理多尺度特征，并提出IoU感知查询选择机制，通过为解码器提供更高质量的初始目标查询进一步提升性能。其精度与速度均优于同期YOLO系列，获得了广泛关注。RT-DETRv2[19]进一步优化了训练策略，包括动态数据增强和便于部署的优化采样算子，使模型性能得到更进一步提升。但由于其采用一对一稀疏监督机制，收敛速度与最终效果受到限制。因此，引入一对多标签分配策略可进一步提升模型性能。

2.3. 辅助训练策略

Co-DETR [33]提出了多种并行的一对多标签分配辅助头训练策略（如ATSS [30]和Faster RCNN [20]），这些策略能够轻松增强端到端检测器中编码器的学习能力。例如，ViT-CoMer [27]与Co-DETR [33]的结合在COCO检测任务上实现了最先进的性能。DAC-DETR [9]、MS-DETR [31]和GroupDETR [4]主要通过向模型的解码器添加一对多监督信息来加速模型的收敛。上述方-

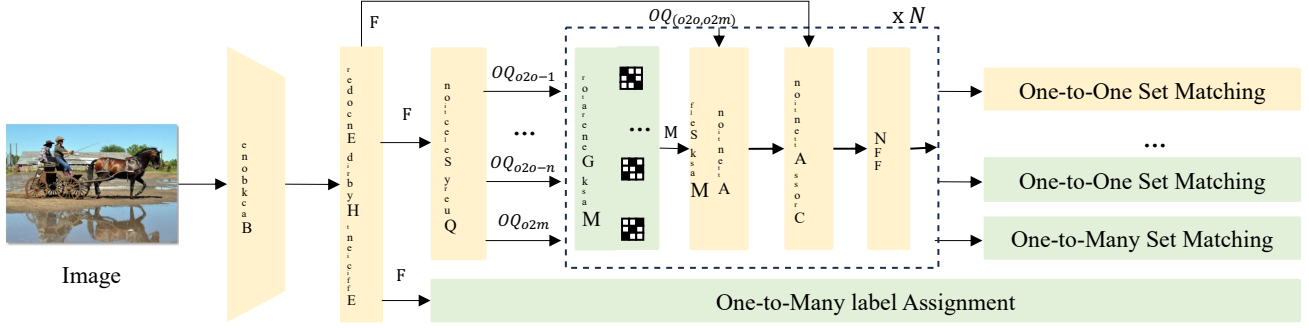


图2. RT-DETRv3架构图。我们保留了RT-DETR的核心架构（黄色高亮部分），并提出了一种新颖的分层解耦密集监督方法（绿色强调部分）。首先，通过引入基于CNN的一对多标签分配辅助分支，增强了编码器的表征能力。其次，为加强解码器的监督力度，我们通过查询选择模块生成多组对象查询(OQ)，并采用随机掩码干扰自注意力机制，有效丰富了正查询样本的分布多样性。此外，为确保多个相关查询聚焦同一目标，我们额外引入了一对多匹配分支作为补充。值得注意的是，这些辅助分支在评估阶段会被移除。

这些方法通过在模型的不同位置添加额外的辅助分支来加速收敛或提升模型性能，但它们并非实时目标检测器。受此启发，我们在RT-DETR[32]的编码器和解码器中引入了多重一对多辅助密集监督模块。这些模块不仅加快了RT-DETR[32]的收敛速度，还提升了其整体性能。由于这些模块仅在训练阶段参与，因此不会影响RT-DETR[32]的推理延迟。

域掩码。这些掩码被应用于自注意力模块，影响查询之间的相关性，从而区分正查询的分配。每组随机掩码都与一个对应的查询配对，如图2中的 $OQ_{o2o-1}, \dots, OQ_{o2o-n}$ 所示。此外，为了确保每个真实标注能匹配到更多高质量查询，我们在解码器中引入了一对多标签分配分支。接下来的章节将详细描述本工作中提出的各个模块。

3. 方法

3.1. 总体架构

RT-DETRv3的整体结构如图2所示。我们保留了RT-DETR[32]的整体框架（黄色高亮部分），并额外引入了提出的分层解耦密集监督方法（绿色高亮部分）。首先，输入图像通过CNN骨干网络（如ResNet[8]）和特征融合模块——即高效混合编码器进行处理，获得多尺度特征 $\{C_3, C_4 \text{ 和 } C_5\}$ 。这些特征随后并行输入基于CNN的一对多辅助分支和基于Transformer的解码器分支。对于基于CNN的一对多辅助分支，我们直接采用现有最先进的密集监督方法（如PP-YOLOE[28]）来协同监督编码器的表征学习。在基于Transformer的解码器分支中，多尺度特征首先被展平并拼接。接着通过查询选择模块从中筛选出top-k特征以生成对象查询。在解码器内部，我们引入了能生成多组随机掩码的掩码生成器——

3.2. RT-DETR概述

RT-DETR [32] 是一种专为物体检测任务设计的实时检测框架。它融合了DETR [3]端到端预测的优势，同时优化了推理速度和检测精度。为实现实时性能，该框架采用轻量级CNN主干网络替换原编码器模块，并设计了高效混合编码器模块以实现特征的高效融合。RT-DETR [32]提出不确定性最小化查询选择模块，通过筛选高置信度特征作为物体查询，降低查询优化难度。随后通过多层解码器中的自注意力、交叉注意力和前馈网络(FFN)模块增强这些查询，最终由MLP层输出预测结果。在训练优化过程中，RT-DETR [32]采用匈牙利匹配算法进行一对一分配，并联合使用L1损失和GIoU损失监督边界框回归，同时采用可变焦点损失(VFL)监督分类任务的学习。

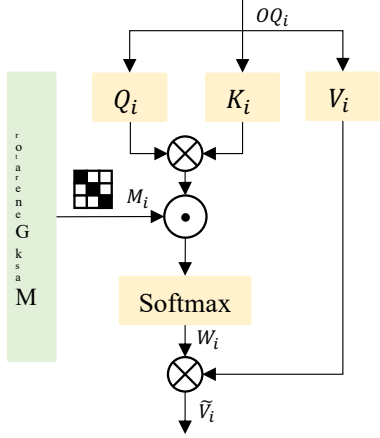


图3. 掩码自注意力模块。\$M_i\$代表与\$i\$-th组对象查询相对应的扰动掩码。⊗表示矩阵乘法，⊙表示逐元素乘法。

3.3. 基于CNN的一对多辅助分支

为缓解解码器一对多集合匹配方案导致编码器输出监督稀疏的问题，我们引入了具有一对多分配策略的辅助检测头，例如PP-YOLOE[28]。该策略能有效增强对编码器的监督，使其具备足够的表征能力以加速模型收敛。具体而言，我们将编码器的输出特征\$\{C_3, C_4\$和\$C_5\}\$直接接入PP-YOLOE检测头。针对一对多匹配算法，我们遵循PP-YOLOE头的配置方案：在训练初期采用ATSS匹配算法，后期切换至TaskAlign匹配算法。对于分类与定位任务的学习，分别选用VFL（Varifocal Loss）和分布式焦点损失（DFL）。其中VFL以IoU分数作为正样本目标，使得高IoU正样本对损失贡献更大，这促使模型在训练过程中更关注高质量样本而非低质量样本。值得注意的是，解码器头部同样采用VFL损失以确保任务定义的一致性。我们将CNN辅助分支的整体损失记为\$L_{aux}\$，其对应损失权重记为\$\alpha\$。

3.4. 基于Transformer的多组自注意力扰动分支

解码器由一系列Transformer块组成，每个块包含自注意力模块、交叉注意力模块以及FFN（前馈网络）模块。最初，查询通过自注意力模块相互交互，以增强或削弱其特征表达——

随后，每个查询通过交叉注意力模块从编码器的输出特征中检索信息来更新自身。最后，前馈网络（FFN）预测与每个查询相对应的目标类别和边界框坐标。然而，RT-DETR采用的一对一集合匹配机制导致了监督信息的稀疏性，最终影响了模型的性能。

为确保与同一目标相关的多个查询有机会参与正样本学习，我们提出了基于掩码自注意力（Mask Self-Attention）的多重自注意力扰动模块。该扰动模块的实现细节如图2所示。首先，通过查询选择模块生成多组对象查询，记为\$OQ_i\$ (\$i=1\dots N\$，其中\$N\$为组数)。相应地，我们使用掩码生成器为每组\$OQ_i\$生成随机扰动掩码\$M_i\$。\$OQ_i\$与\$M_i\$共同输入掩码自注意力模块，最终得到扰动融合后的特征。

掩码自注意力模块的具体实现如图3所示，首先对\$OQ_i\$进行线性投影得到\$Q_i\$、\$K_i\$和\$V_i\$。接着，将\$Q_i\$与\$K_i\$相乘计算注意力权重，该权重再与\$M_i\$相乘并通过softmax函数处理，生成扰动后的注意力权重。最后，将此扰动权重与\$V_i\$相乘，得到融合结果\$\tilde{V}_i\$。该过程可表示为：

$$Q_i, K_i, V_i = \text{Linear}(OQ_i) \quad (1)$$

$$W_i = \text{Softmax}(M_i(Q_i K_i^T)) \quad (2)$$

$$\tilde{V}_i = W_i V_i \quad (3)$$

引入多组随机扰动使查询特征多样化，使得与同一目标相关联的多个相关查询有机会被分配为正样本查询，从而丰富了监督信息。训练过程中，多组对象查询被拼接后输入单一解码器分支，实现参数共享并提升训练效率。损失计算与标签分配方案保持与RT-DETR一致。我们将\$i\$-th集合的损失记为\$L_{o2o}^i\$，\$N\$组扰动集合的总损失计算如下：

$$L_{o2o} = \frac{1}{N} \sum_{i=1}^N L_{o2o}^i \quad (4)$$

对应的损失权重记为\$\beta\$。

3.5. 基于Transformer的一对多密集监督分支

为了最大化多组自注意力扰动分支的效益，我们引入了一个额外的密集

Model	Backbone	#Params(M)	GFLOPs	Latency (ms)	AP^{val}_{1x}	AP^{val}_{3x}	AP^{val}_{6x}
RT-DETR [32]	R18	20	60	4.6	38.7	44.5	46.5
RT-DETRv2 [19]					39.8	44.9	46.7/47.9 [†]
RT-DETRv3 (ours)					41.5	46.1	48.1/48.7[†]
RT-DETR [32]	R34	31	92	6.3	42.8	47.5	48.9
RT-DETRv2 [19]					43.0	47.2	49.0/49.9 [†]
RT-DETRv3 (ours)					44.7	48.6	49.9/50.1[†]
RT-DETR [32]	R50	42	136	9.2	48.9	52.2	53.1
RT-DETRv2 [19]					-	-	53.4
RT-DETRv3 (ours)					50.2	53.0	53.4
RT-DETR [32]	R101	76	259	13.5	50.2	53.5	54.3
RT-DETRv2 [19]					-	-	54.3
RT-DETRv3 (ours)					51.3	54.2	54.6

表1. RT-DETR系列在COCO val2017目标检测任务上的性能对比。R18、R34、R50和R101分别代表ResNet-18、ResNet-34、ResNet-50和ResNet-101网络架构。1x、3x和6x分别对应12、36和72轮次训练。[†]表示进行了120轮次训练。

解码器中采用共享权重的监督分支。这确保了更高质量的查询与每个真实标注相匹配。具体而言，我们利用查询选择模块生成一组独特的对象查询。在样本匹配阶段，通过将训练标签复制 m 倍（默认值为4）生成一个增强的目标集。随后，该增强集将与查询预测进行匹配。损失计算保持与原始检测损失一致，我们指定 L_{o2m} 作为该分支的损失函数，损失权重为 γ 。

3.6. 总损失

综上所述，我们提出的方法的总体损失函数如下：

$$L = \alpha L_{aux} + \beta L_{o2o} + \gamma L_{o2m} \quad (5)$$

其中 L_{aux} 负责对编码器进行密集监督， L_{o2o} 在保留端到端预测特性的同时，为解码器丰富了一对一监督信息，而 L_{o2m} 则为解码器提供了一对多的密集监督。默认情况下，损失权重 α 、 β 和 γ 均设为1。

4. 实验

4.1. 数据集与评估指标

我们选用MS COCO 2017 [15]目标检测数据集作为方法的评估基准。该数据集包含115k训练图像和5k测试图像。

我们采用了与RT-DETR[32]方法相同的评估指标AP。在收敛效率、推理速度和有效性方面，我们将RT-DETRv3与其他实时目标检测器进行了比较，这些检测器包括基于Transformer和基于CNN的实时目标检测器。此外，我们还对本文提到的模块进行了消融实验。所有实验细节和结果将在以下部分详细阐述。

4.2. 实现细节

我们将提出的分层密集监督分支集成到了RT-DETR[32]框架中。基于CNN的密集监督辅助分支直接采用了PP-YOLOE头部结构，其样本匹配策略、损失计算及其他所有配置均与PP-YOLOE[28]保持一致。我们复用RT-DETR[32]的解码器结构作为主分支，并额外添加了三组参数共享的自注意力扰动分支。样本匹配方式与主分支一致，采用匈牙利匹配算法。同时新增了一个参数共享的一对多匹配分支，默认每个真实标注框匹配四个目标查询，共设置300个目标查询。优化器采用集成权重衰减因子0.0001的AdamW，确保其他所有训练配置——包括数据增强与预训练——严格遵循RT-DETR[32]方案。较小骨干网络（R18、R34）采用10倍（120轮）训练周期，较大骨干网络（R50、R101）采用6倍（72轮）训练周期。我们使用四块NVIDIA

Model	#Epochs	#Params (M)	GFLOPs	Latency (ms)	AP^{val}
YOLOv6-3.0-S [13]	300	18.5	45.3	3.4	44.3
Gold-YOLO-S [23]	300	21.5	46.0	3.8	45.4
YOLO-MS-S [5]	300	8.1	31.2	10.1	46.2
YOLOv8-S [11]	500	11.2	28.6	7.1	46.2
YOLOv9-S [26]	500	7.1	26.4	-	46.7
YOLOV10-S [22]	500	7.2	21.6	2.5	46.3
RT-DETRv3-R18 (ours)	120	20	60	4.6	48.7
YOLOv6-3.0-M [13]	300	34.9	85.8	5.6	49.1
Gold-YOLO-M [23]	300	41.3	87.5	6.4	49.8
YOLO-MS [5]	300	22.2	80.2	12.4	51.0
YOLOv8-M [11]	500	25.9	78.9	9.5	50.6
YOLOv9-M [26]	500	20.0	76.3	-	51.1
YOLOV10-M [22]	500	15.4	59.1	4.7	51.1
RT-DETRv3-R34 (ours)	120	31	92	6.3	50.1
RT-DETRv3-R50m (ours)	72	36	100	6.89	51.7
Gold-YOLO-L [23]	300	75.1	151.7	9.0	51.8
YOLOv5-X [12]	300	86	205	23.3	50.7
PPYOLOE-L [28]	300	52	110	10.6	51.4
YOLOv6-L [13]	300	59	150	10.1	52.8
YOLOv7-L	300	36	104	18.2	51.2
YOLOV8-L [11]	500	43	165	14.1	52.9
YOLOv9-C [26]	500	25.3	102.1	10.57	52.5
YOLOV10-L [22]	500	24.4	120.3	7.28	53.2
RT-DETRv3-R50 (ours)	72	42	136	9.2	53.4
YOLOv8-X [11]	500	68.2	257.8	16.9	53.9
YOLOv10-X [22]	500	29.5	160.4	10.7	54.4
RT-DETRv3-R101 (ours)	72	76	259	13.5	54.6

表2. 与基于CNN的实时目标检测器在COCO val2017上的对比。

使用A100 GPU以64的批量大小训练我们提出的方法。此外，所有模型的延迟均在T4 GPU上采用TensorRT FP 16进行测试，遵循[32]的方法。我们观察到，与大多数采用更长训练周期的检测器相比，RT-DETRv3仅需72个周期即可达到卓越的准确度。

4.3. 与基于Transformer的实时目标检测器对比

推理速度与算法性能。基于Transformer架构的实时目标检测器主要以RT-DETR系列为代表。表1展示了我们的方法与RT-DETR系列的对比结果。我们的方法在{v*}方面均优于

RT-DETR [32]和RT-DETRv2 [19]在不同骨干网络上的表现。具体而言，与RT-DETR [32]相比，在6倍训练周期下，我们的方法在R18、R34、R50和R101骨干网络上分别实现了1.6%、1.0%、0.3%和0.3%的性能提升。相较于RT-DETRv2 [19]，我们在6倍/10倍训练周期下评估了R18和R34骨干网络，我们的方法分别带来了1.4%/0.8%和0.9%/0.2%的改进。此外，由于我们提出的辅助密集监督分支仅用于训练阶段，我们的方法在推理速度上与RT-DETR [32]和RT-DETRv2 [19]保持一致。

收敛速度。我们的方法基于RT-DETR[32]框架，通过融入基于CNN和

Method	Extra data	Epochs	AP^{val}
RT-DETRv3-R50	x	51	53.4
	x	72	52.9 (-0.5)
	✓	51	54.2
	✓	72	54.8 (+0.6)
RT-DETRv3-R101	x	51	54.6
	x	72	54.2 (-0.4)
	✓	51	54.7
	✓	72	55.4 (+0.7)

表3. 过拟合分析。额外数据代表Object365数据集[21]。

基于transformer的一对多密集监督方法，不仅提升了模型性能，还加速了收敛速度。我们通过大量实验验证了该方法的有效性。表1展示了RT-DETRv3、RT-DETR[32]和RT-DETRv2[19]在不同训练周期下的对比分析，清晰表明我们的方法在任何训练周期下都具备更快的收敛速度，且仅需一半训练轮次即可达到相当性能水平。

过拟合分析。如图4所示，我们注意到随着模型规模增大，RT-DETRv3容易出现过拟合现象。我们认为这可能是训练数据集规模与模型规模不匹配所致。如表3所示，当我们增加额外训练数据时，RT-DETRv3的性能会随着训练轮次增加而持续提升，且在相同训练轮次下表现优于未添加额外数据的模型。

4.4. 与基于CNN的实时目标检测器比较

推理速度与算法性能。我们将RT-DETRv3的端到端速度和精度与当前先进的基于CNN的实时目标检测方法进行了比较。根据推理速度，将模型分为小、中、大规模。在相近的推理性能条件下，将RT-DETRv3与YOLOv6-3.0[13]、Gold-YOLO[23]、YOLO-MS[5]、YOLOv8[11]、YOLOv9[26]、YOLOv10[22]等前沿算法进行对比。如表2所示，在小规模模型中，RT-DETRv3-R18方法分别以4.4%、3.3%、2.5%、2.5%、2.0%和2.4%的优势超越YOLOv6-3.0-S、Gold-YOLO-S、YOLO-MS-S、YOLOv8-S、YOLOv9-S和YOLOv10-S。在中规模模型中，RT-DETRv3相较YOLOv6-3.0-M、Gold-YOLO-M同样展现出更优性能。

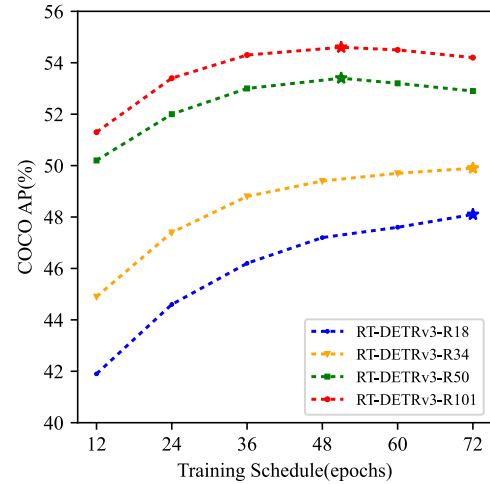


图4. RT-DETRv3在不同模型尺寸下的收敛曲线。★代表最佳平均精度(AP)。

YOLO-MS-M、YOLOv8-M、YOLOv9-M和YOLOv10-M。在大规模模型上，我们的方法持续超越基于CNN的实时目标检测器。例如，我们的RT-DETRv3-R101可实现54.6 AP，优于YOLOv10-X。但由于我们尚未针对轻量化部署优化RT-DETRv3检测器的整体框架，RT-DETRv3的推理效率仍有进一步提升空间。

收敛速度。如表2所示，我们欣喜地发现，在实现卓越性能的同时，我们的RT-DETRv3能将训练周期缩短至基于CNN的实时检测器的60%甚至更少。

4.5. 消融研究。

设置。我们以RT-DETR[32]为基线进行消融实验，依次整合基于辅助CNN的一对多标签分配分支、基于辅助Transformer的一对多标签分配分支以及多组自注意力扰动模块，验证所提方法的影响。实验采用ResNet18作为主干网络，批量大小为64，使用四块NVIDIA A100 GPU，其余配置与RT-DETR[32]保持一致。

组件消融实验。我们进行了消融实验以评估本文提出的各个模块。如表4所示，每个模块都显著提升了模型性能。例如，在RT-DETR[32]基础上添加O2M-T模块后，模型性能较基线提升了1.0%。当所有提出的模块被集成到RT-DETR中进行算法优化时，模型性能提升了1.6%。

Method	O2M-C	O2M-T	MGSA	AP^{val}
RT-DETR	x	x	x	46.5
	✓	x	x	47.4
	x	✓	x	47.5
	x	x	✓	47.5
	✓	✓	✓	48.1

表4. 关键组件的消融研究。O2M-C代表基于CNN的一对多辅助分支，O2M-T指基于transformer的一对多密集监督分支，MGSA表示基于transformer的多组自注意力扰动分支。

Number	AP^{val}	AP_{50}^{val}
2	47.9	65.4
3	48.1	65.6
4	48.0	65.3

表5. 自注意力扰动分支数量的消融研究。

自注意力扰动分支数量。为验证自注意力扰动分支数量对RT-DETRv3性能的影响，我们采用RT-DETRv3-R18模型进行消融实验，分别设置分支数为2、3、4，同时保持其他配置不变。如表5所示，当分支数设为3时，模型以48.1的AP值达到最佳性能。减少分支数会降低监督信号的丰富性，导致性能下降；而过度增加分支数则会提升模型学习难度，却未能带来显著的性能提升。

5. 结论

本文提出了一种基于Transformer的实时目标检测算法RT-DETRv3。该算法在RT-DETR基础上，通过引入多重密集正样本辅助监督模块进行改进。这些模块对RT-DETR编码器和解码器的特定特征实施一对多目标监督，从而加速算法收敛并提升性能。需要特别说明的是，这些模块仅用于训练阶段。我们在COCO目标检测基准上验证了算法的有效性，实验表明相比其他实时检测器，本算法取得了更优结果。希望这项工作能为基于Transformer的实时目标检测领域的研究者与开发者带来启发。

参考文献

[1] Alexey Bochkovskiy、Chien-Yao Wang 和 Hong-Yuan Mark Liao. Yolov4: 目标检测的最佳速度与精度。arXiv preprint arXiv:2004.10934, 2020年。1, 2 [2] Nicolas Carion、Francisco Massa、Gabriel Synnaeve、Nicolas Usunier、Alexander Kirillov 和 Sergey Zagoruyko。基于Transformer的端到端目标检测。载于European conference on computer vision, 第213–229页。Springer, 2020年。2 [3] Nicolas Carion、Francisco Massa、Gabriel Synnaeve、Nicolas Usunier、Alexander Kirillov 和 Sergey Zagoruyko。基于Transformer的端到端目标检测。载于European conference on computer vision, 第213–229页。Springer, 2020年。3 [4] 陈强、陈晓康、王健、张山、姚坤、冯浩成、韩俊宇、丁二锐、曾刚和王京东。Group DETR: 通过分组一对多分配实现快速DETR训练。载于Proceedings of the IEEE/CVF International Conference on Computer Vision, 第6633–6642页, 2023年。2 [5] 陈玉明、袁新斌、吴瑞琪、王家宝、侯启斌和程明明。YOLO-MS: 重新思考实时目标检测中的多尺度表征学习。arXiv preprint arXiv:2308.05480, 2023年。6, 7 [6] 丁晓晗、张翔宇、马宁宁、韩军功、丁贵广和孙剑。RepVGG: 让VGG风格卷积网络重焕光彩。载于Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 第13733–13742页, 2021年。2 [7] 葛政、刘松涛、王峰、李泽明和孙剑。YOLOX: 2021年超越YOLO系列。arXiv preprint arXiv:2107.08430, 2021年。2 [8] 何恺明、张翔宇、任少卿和孙剑。深度残差学习在图像识别中的应用。载于Proceedings of the IEEE conference on computer vision and pattern recognition, 第770–778页, 2016年。3 [9] 胡正东、孙逸凡、王京东和杨毅。DAC-DETR: 分治注意力层。Advances in Neural Information Processing Systems, 36卷, 2024年。2 [10] 黄鑫、王欣欣、吕文宇、白雪莹、龙翔、邓凯鹏、党青青、韩树民、刘奇文、胡晓光等。PP-YOLOv2: 实用目标检测器。arXiv preprint arXiv:2104.10419, 2021年。2 [11] Glenn Jocher、Ayush Chaurasia 和 Qiu Jing。Ultralytics YOLO, 2023年1月。1, 2, 6, 7 [12] Glenn Jocher、Ayush Chaurasia、Alex Stoken、Jirka Borovec、Yonghye Kwon、Kalen Michael、Jiacong Fang、Colin Wong、Zeng Yifu、Diego Montes等。ultralytics/yolov5: v6.2-YOLOv5分类模型, Apple M1支持, 可复现性, ClearML与Deci.ai集成。Zenodo, 2022年。1, 2, 6 [13] 李楚怡、李露露、姜洪亮、翁凯恒、耿一飞、李亮、柯再丹、李庆元、程梦、聂伟强等。YOLOv6: 面向工业应用的单阶段目标检测框架。arXiv preprint arXiv:2209.02976, 2022年。1, 2, 6, 7 [14] 李峰、张浩、刘世龙、郭健、Ni Lionel M 和张磊。DN-DETR: 通过查询去噪加速DETR训练。载于Proceedings of the IEEE/CVF

conference on computer vision and pattern recognition, 第13619–13627页, 2022年。2 [15] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár and C Lawrence Zitnick。Microsoft COCO: 上下文中的常见物体。载于 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, 第740–755页。Springer, 2014年。5 [16] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu and Lei Zhang。DAB-DETR: 动态锚框是DETR更优的查询。arXiv preprint arXiv:2201.12329, 2022年。2 [17] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi and Jiaya Jia。用于实例分割的路径聚合网络。载于 *Proceedings of the IEEE conference on computer vision and pattern recognition*, 第8759–8768页, 2018年。2 [18] Xiang Long, Kaipeng Deng, Guanzhong Wang, Yang Zhang, Qingqing Dang, Yuan Gao, Hui Shen, Jianguo Ren, Shumin Han, Errui Ding等。PP-YOLO: 一种高效且有效的目标检测器实现。arXiv preprint arXiv:2007.12099, 2020年。2 [19] Wenyu Lv, Yian Zhao, Qinyao Chang, Kui Huang, Guanzhong Wang and Yi Liu。RT-DETRv2: 通过免费技巧包改进的实时检测Transformer基线。arXiv preprint arXiv:2407.17140, 2024年。2, 5, 6, 7 [20] Shaoqing Ren, Kaiming He, Ross Girshick and Jian Sun。Faster R-CNN: 利用区域提议网络实现实时目标检测。Advances in neural information processing systems, 28, 2015年。2 [21] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li and Jian Sun。Objects365: 用于目标检测的大规模高质量数据集。载于 *Proceedings of the IEEE/CVF international conference on computer vision*, 第8430–8439页, 2019年。7 [22] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han and Guiguang Ding。YOLOv10: 实时端到端目标检测。arXiv preprint arXiv:2405.14458, 2024年。1, 6, 7 [23] Chengcheng Wang, Wei He, Ying Nie, Jianyuan Guo, Chuanjian Liu, Yunhe Wang and Kai Han。Gold-YOLO: 通过收集-分发机制实现的高效目标检测器。Advances in Neural Information Processing Systems, 36, 2024年。6, 7 [24] Chien-Yao Wang, Alexey Bochkovskiy and Hong-Yuan Mark Liao。YOLOv7: 可训练免费技巧包为实时目标检测器设定新标杆。载于 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 第7464–7475页, 2023年。1, 2 [25] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh and I-Hau Yeh。CSPNet: 一种能增强CNN学习能力的新骨干网络。载于 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 第390–391页, 2020年。2 [26] Chien-Yao Wang, I-Hau Yeh and Hong-Yuan Mark Liao。YOLOv9: 利用可编程梯度信息学习你想学的内容。arXiv preprint arXiv:2402.13616, 2024年。1, 2, 6, 7

[27] 夏春龙, 王新亮, 吕峰, 郝鑫, 石一峰。Vit-comer: 用于密集预测的卷积多尺度特征交互视觉Transformer。载于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 第5493–5502页, 2024年。2 [28] 徐尚亮, 王欣欣, 吕文宇, 常钦尧, 崔程, 邓凯鹏, 王冠中, 党青青, 魏胜宇, 杜宇宇等。PP-YOLOE: YOLO的进化版本。arXiv preprint arXiv:2203.16250, 2022年。3, 4, 5, 6 [29] 张浩, 李峰, 刘世龙, 张磊, 苏航, 朱军, 倪立昂, 沈向洋。DINO: 基于改进去噪锚框的端到端目标检测DETR模型。arXiv preprint arXiv:2203.03605, 2022年。2 [30] 张世峰, 迟程, 姚永强, 雷震, 李世章。通过自适应训练样本选择弥合基于锚与无锚检测的差距。载于 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 第9759–9768页, 2020年。2 [31] 赵楚阳, 孙一凡, 王文浩, 陈强, 丁二瑞, 杨毅, 王京东。MS-DETR: 混合监督下的高效DETR训练。载于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 第17027–17036页, 2024年。2 [32] 赵一安, 吕文宇, 徐尚亮, 魏金满, 王冠中, 党青青, 刘毅, 陈杰。实时目标检测中DETR超越YOLO。载于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 第16965–16974页, 2024年。2, 3, 5, 6, 7 [33] 宗卓凡, 宋广录, 刘宇。协作混合分配训练的DETR模型。载于 *Proceedings of the IEEE/CVF international conference on computer vision*, 第6748–6758页, 2023年。2