DETRs Beat YOLOs on Real-time Object Detection

摘要

本文介绍了第一个基于 Transformer 的端到端的实时的目标检测框架 RT-DETR。传统基于 CNN 的 YOLO 模型的速度性能受限于非极大值抑制后处理，而基于 Transformer 的目标检测框架则受限于参数量和计算量，导致了当前的研究对基于 Transformer 的实时目标检测没有得到一个良好的探索。RT-DETR 改进了上述所存在的问题，实现了速度和检测精度的双提升。学生在人车目标检测数据集上验证了 RT-DETR 的有效性，并根据其存在的不足，提出了特征融合敏感的特征融合模块：FGConcat。实验结果表明所提出方法的有效性，其对小目标具有更有效的检出率。

关键词：目标检测；深度学习；RT-DETR

1 引言

目前 YOLO 是最流行的实时目标检测框架，但是它的速度和精度受到了后处理步骤中的非极大值抑制（NMS）的负面影响。而目前端到端的基于 Transformer 架构的目标检测器（DETRs）虽然可以在目标检测过程中消除非极大值抑制，但是基于 Transformer 架构的目标检测框架也有着高计算量开销的缺陷，所以限制了 DETRs 在实时检测领域的运用。基于以上所述的背景，研发一个基于 Transformer 的实时端到端的目标检测框架将在目标检测领域具有指导性的价值意义，所以作者提出了 RT-DETR [7]。目标检测在实际生活中有着非常广泛的应用，它是许多下游任务的基础工作。比如无人汽车，智能驾驶等应用场景，汽车需要根据周围的环境来检测其他汽车、行人等目标，以此来达到避让的目的。

2 相关工作

2.1 实时目标检测器

YOLOv1 是第一个基于 CNN 的单级目标检测器，实现真正的实时目标检测。经过多年的不断发展，YOLO 检测器已经超越了其他单级物体检测器，成为实时物体检测器的代名词。YOLO 检测器可以分为两类：基于 anchor 的和 anchor-free，它们在速度之间实现了合理的权衡和准确性，广泛应用于各种实际场景。这些先进的实时检测器会产生大量重叠的框，需要 NMS 后处理，这会降低其速度。

2.2 端到端目标检测器

端到端对象检测器以其简化的管道而闻名。卡里昂等人。[1] 首先提出了基于 Transformer 的端到端检测器 DETR，由于其独特的特点而引起了广泛的关注。特别是，DETR 消除了手工制作的锚点和 NMS 组件。相反，它采用二分匹配并直接预测一对一的对象集。尽管 DETR 具有明显的优势，但它也存在一些问题：训练收敛速度慢、计算成本高、查询难以优化。已经提出了许多 DETR 变体来解决这些问题。**加速融合**。 Deformable-DETR [8] 通过提高注意力机制的效率来加速多尺度特征的训练收敛。DAB-DETR [5] 和 DN-DETR [3] 通过引入迭代细化方案和去噪训练进一步提高性能。Group-DETR [2] 引入了分组一对多分配。**降低计算成本**。高效 DETR 和稀疏 DETR 过减少编码器和解码器层的数量或更新查询的数量来降低计算成本。Lite DETR 通过以交错的方式降低低级特征的更新频率来提高编码器的效率。**优化初始化的查询向量**。Conditional DETR 和 Anchor DETR 降低了查询的优化难度。朱等人。[8] 提出了两阶段 DETR 的查询选择，DINO 提出了混合查询选择以帮助更好地初始化查询。当前的 DETR 仍然是计算密集型的，并且不是为实时检测而设计的。我们的 RT-DETR 积极探索降低计算成本并尝试优化查询初始化，超越最先进的实时检测器。

3 本文方法

本文首先通过实验来分析非极大值抑制对目标检测器的影响，然后针对所发现的不足来进行改进，接下来将逐步来阐述本文的方法。

3.1 关于非极大值抑制的速度分析

非极大值抑制是目标检测中广泛使用的后处理算法，用于消除重叠的输出框。非极大值抑制算法中需要两个阈值：置信度阈值和 IoU 阈值。具体来说，得分低于置信度阈值的框被直接过滤掉，每当任意两个框的 IoU 超过 IoU 阈值时，得分较低的框将被丢弃。算法运行过程中将迭代执行此过程，直到处理完每个类别的所有框为止。因此，非极大值抑制的执行时间主要取决于框的数量和这两个阈值。为了验证这一观察结果，论文作者利用 YOLOv5（基于锚框）和 YOLOv8（无锚框）进行分析。

作者首先计算在同一输入上使用不同置信度阈值过滤输出框后剩余的框数。他们采样 0.001 到 0.25 之间的值作为置信度阈值，统计两个检测器剩余框的数量，并将其绘制在条形图上，这直观地反映了非极大值抑制对超参数的敏感度，如图 1所示。随着置信度阈值的增加，更多预测框被过滤掉，剩余需要计算 IoU 的框数量减少，从而减少了非极大值抑制的执行时间。

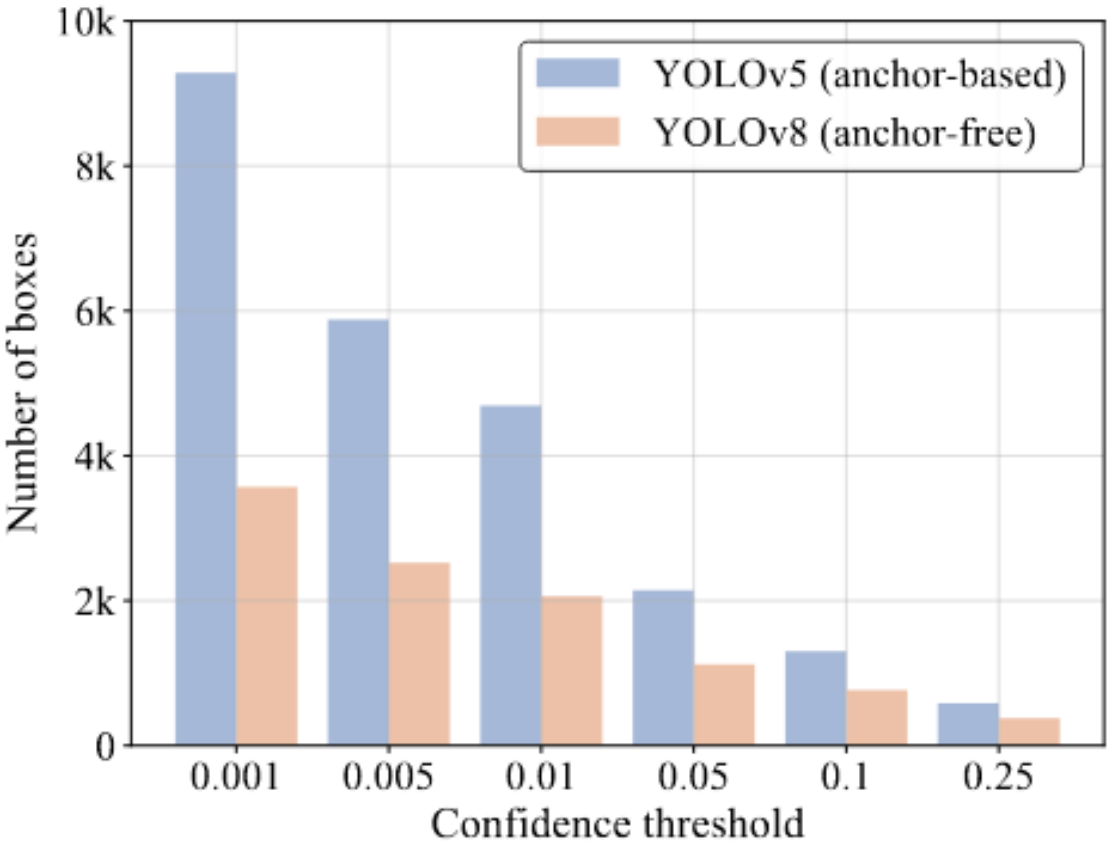

图 1. 不同置信度阈值下的框数

3.2 本文方法概述

接下来对本文将要复现的工作进行概述，作者所提出的模型的架构图如图 2所示。

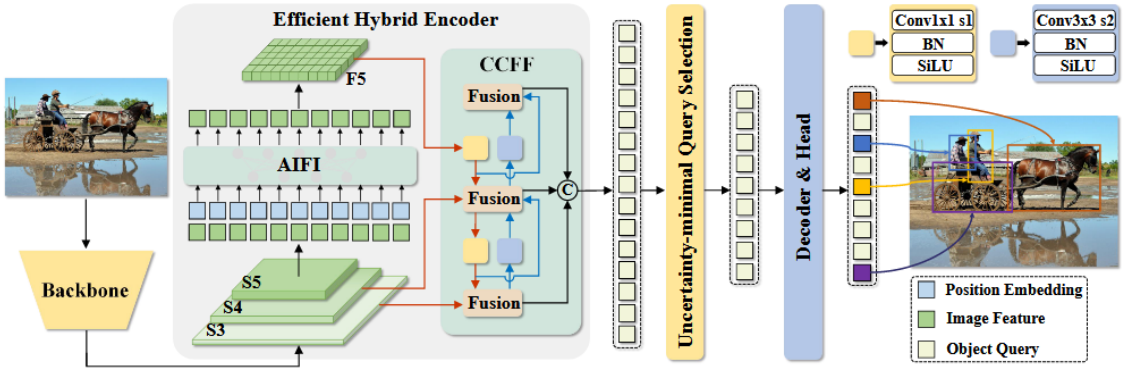

图 2. RT-DETR 架构图。我们将主干最后三个阶段的特征输入编码器。高效的混合编码器通过基于注意力的尺度内特征交互（AIFI）和基于 CNN 的跨尺度特征融合（CCFF）将多尺度特征转换为图像特征序列。然后，不确定性最小查询选择选择固定数量的编码器特征作为解码器的初始对象查询。最后，具有辅助预测头的解码器迭代优化对象查询以生成类别和框。

RT-DETR 由骨干网络、高效混合编码器和带有辅助预测头的 Transformer 解码器组成。具体来说，我们将主干网最后三个阶段 S3、S4、S5 的特征输入到编码器中。高效的混合编码器通过尺度内特征交互和跨尺度特征融合将多尺度特征转换为图像特征序列。随后，采用不

确定性最小查询选择来选择固定数量的编码器特征作为解码器的初始对象查询。最后，具有辅助预测头的解码器迭代优化对象查询以生成类别和框。

3.3 高效的混合编码器

计算瓶颈分析。多尺度特征的引入加速了训练收敛并提高了性能 [8]。然而，虽然可变形注意力降低了计算成本，但急剧增加的序列长度仍然导致编码器成为计算瓶颈。据 lin 等人报道 [4]，编码器占 GFLOP 的 49%，但在 Deformable-DETR 中仅贡献 AP 的 11%。为了克服这个瓶颈，我们首先分析多尺度 Transformer 编码器中存在的计算冗余。直观上，包含对象丰富语义信息的高级特征是从低级特征中提取的，使得对级联的多尺度特征进行特征交互变得多余。因此，我们设计了一组具有不同类型编码器的变体，以证明同时进行的尺度内和跨尺度特征交互是低效的，图 3。特别地，我们使用具有较小尺寸数据读取器的 DINO-Deformable-R50 和 RT-DETR 中使用的较轻的解码器进行实验，首先去除 DINO-Deformable-R50 中的多尺度 Transformer 编码器作为变体 A。然后，插入不同类型的编码器以产生基于 A 的一系列变体，详细阐述如下：

- A → B：变体 B 将单尺度 Transformer 编码器插入到 A 中，该编码器使用一层 Transformer 块。多尺度特征共享编码器以进行尺度内特征交互，然后连接作为输出。
- B → C：变体 C 在 B 的基础上引入跨尺度特征融合，并将连接后的特征输入到多尺度 Transformer 编码器中，以同时执行尺度内和跨尺度特征交互。
- C → D：变体 D 通过利用前者的单尺度 Transformer 编码器和后者的 PANet 式 [21] 结构来解耦尺度内交互和跨尺度融合。
- D → E：变体 E 在 D 的基础上增强了尺度内交互和跨尺度融合，采用了我们设计的高效混合编码器。

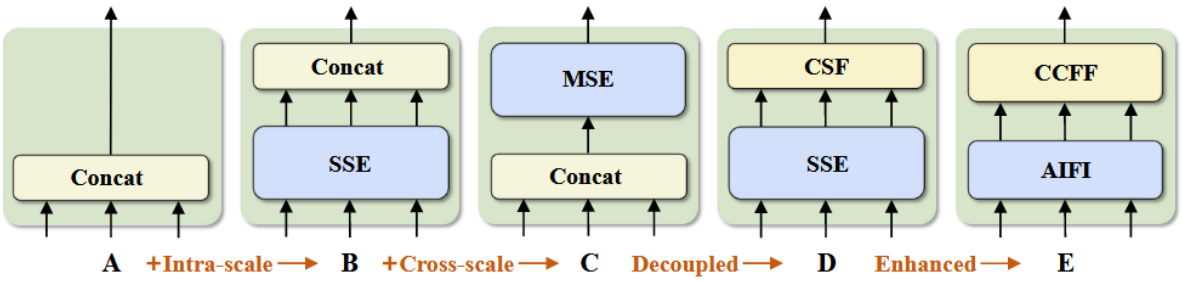

图 3. 每个变体的编码器结构。SSE 代表单尺度 Transformer 编码器，MSE 代表多尺度 Transformer 编码器，CSF 代表跨尺度融合。AIFI 和 CCFF 是作者论文中的混合编码器中设计的两个模块。

基于上述分析，作者重新思考编码器的结构，并提出了一种高效的混合编码器，由两个模块组成，即基于注意力的尺度内特征交互（AIFI）和基于 CNN 的跨尺度特征融合（CCFF）。具体来说，AIFI 通过仅在 S5 上使用单尺度 Transformer 编码器进行尺度内交互，进一步降低了基于变体 D 的计算成本。原因是，将自注意力操作应用于具有更丰富语义概念的高级特

征，捕获了概念实体之间的联系，这有利于后续模块对对象的定位和识别。然而，由于缺乏语义概念以及与高层特征交互存在重复和混淆的风险，低层特征的尺度内交互是不必要的。所以 AIFI 通过仅在 S5 上使用单尺度 Transformer 编码器进行尺度内交互。

3.4 不确定性最小化查询选择

为了降低 DETR 中优化对象查询的难度，后续的几项工作 [39,41,42] 提出了查询选择方案，其共同点是它们使用置信度分数从编码器中选择前 K 个特征来初始化对象查询（或者只是位置查询）。置信度分数表示该特征包含前景对象的可能性。然而，检测器需要同时对对象的类别和位置进行建模，这两者都决定了特征的质量。因此，该特征的性能得分是一个与分类和定位共同相关的潜在变量。根据分析，当前的查询选择导致所选特征具有相当大的不确定性，导致解码器的初始化不理想并阻碍检测器的性能。

为了解决这个问题，作者提出了不确定性最小查询选择方案，该方案显式地构造和优化认知不确定性来对编码器特征的联合潜在变量进行建模，从而为解码器提供高质量的查询。具体来说，特征不确定性 U 定义为等式中定位 P 和分类 C 的预测分布之间的差异。(2)。为了最大限度地减少查询的不确定性，我们将不确定性集成到损失函数中，以进行基于梯度的优化，损失函数如下列的表达式所示：

$$\mathcal{U}(\hat{\mathcal{X}}) = \|\mathcal{P}(\hat{\mathcal{X}}) - \mathcal{C}(\hat{\mathcal{X}})\|, \quad \hat{\mathcal{X}} \in \mathbb{R}^D$$
$$\mathcal{L}(\hat{\mathcal{X}}, \hat{\mathcal{Y}}, \mathcal{Y}) = \mathcal{L}_{box}(\hat{b}, b) + \mathcal{L}_{cls}(\mathcal{U}(\hat{\mathcal{X}}), \hat{c}, c)$$

4 复现细节

4.1 与已有开源代码对比

与已开源的代码相比，本文通过引入了其他的数据增强方式，以及细粒度特征融合 FG-Concat 模块，从而提出了特征融合敏感的目标检测算法。复现的该论文强调实时性和精度的双提升，故学生将该模型改进后应用到人车目标检测竞赛中，并取得了第二名的成绩。改进后的模型在目标遮挡、以及小目标的检测效果上取得了更佳的性能。为了更直观的表现该竞赛的难度，在后面的章节将进一步对数据进行可视化，并给出目标检测结果的效果图。

4.2 数据处理

4.2.1 数据集介绍

本赛题的数据集共包含 3000 张分辨率统一为 1280×720 行车记录仪拍摄的 JPEG 编码照片，其中包含 2600 张人工标注的两点 anchor box 标签，400 张用于测试集来评价参赛团队的模型性能。标签包括以下人和车的类别共 22 种。其中行人包括普通行人、3D 假人、坐着的人、骑车的人；车辆包括两厢车、三厢车、小型客车、小货车、皮卡车、轻卡、厢式货车、牵引车、水泥车、工程车辆、校车、中小型客车、大型单层客车、小型电动车、摩托车、自行车、三轮车以及其它特殊车辆。

4.2.2　数据集可视化

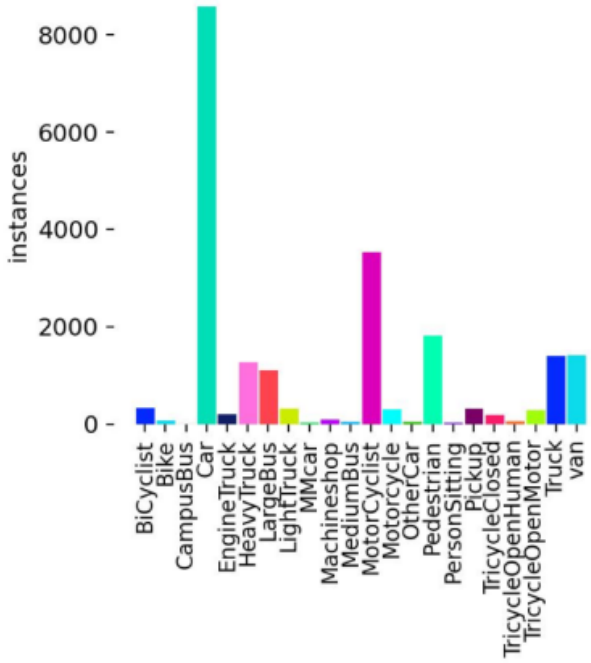

图 4. 数据集实例分布图

图 4对数据集中的目标进行实例统计。其中 y 轴 instances 表示实例的个数。由图中可以看出，数据集存在严重的类别不均衡问题。比如 Bike、CampusBus、MMcar 等类别的实例过于稀少。这将会影响后续模型的性能，导致后续模型对这些类别的检出率低下。

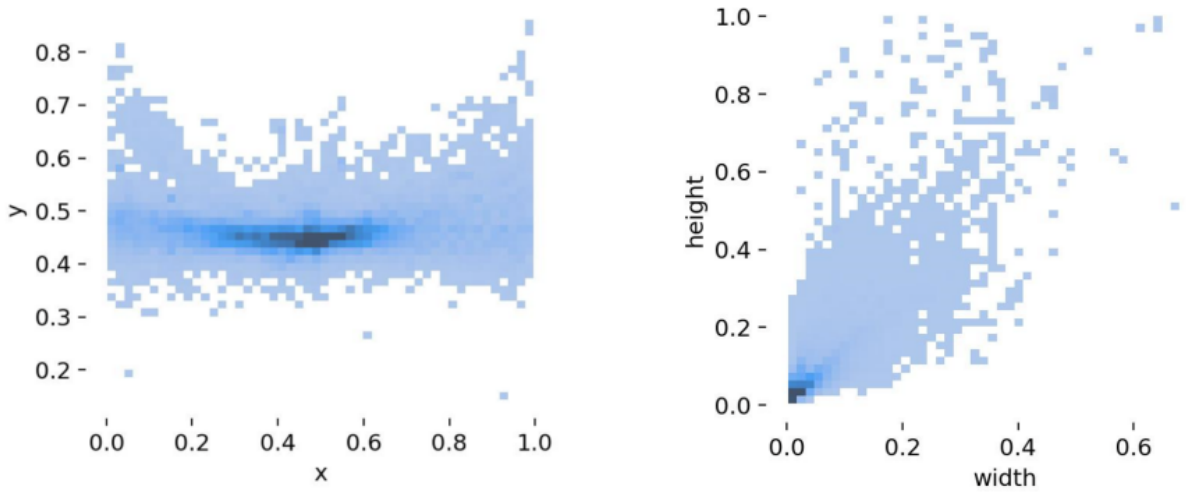

图 5. 实例的中心点分布和宽高分布图

图 5展示了数据集中实例的中心点分布图和实例的宽高分布图，该可视化图是经过归一化的。x 表示了实例在图像中宽度的位置，y 表示了实例在图像中高度的位置。width 表示了实例的宽度占据了图像宽度的百分比，height 表示了实例的高度占据了图像高度的百分比。由图 5可以发现，数据集中实例集中分布在图像在中间，并且实例的高度和宽度都严重偏小，这将对后续模型的训练产生巨大的挑战。

图 6. 目标存在遮挡且目标极小

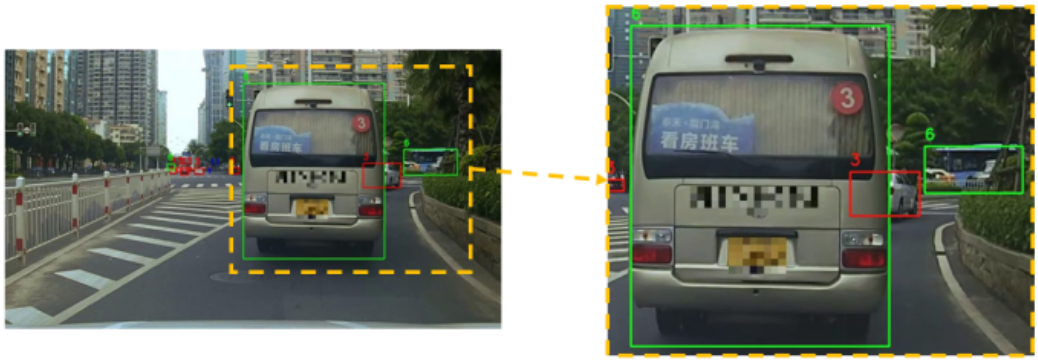

图 7. 目标之间存在严重的遮挡

图 6和图 7是根据赛方提供的标签数据集对部分图像进行锚框可视化。通过可视化锚框后，可以直观的看出该赛题存在的挑战性。

4.3 数据增强

为了缓解数据样本过少的问题，在进行模型训练时，需要对其进行数据增强操作。考虑到篇幅问题，以下简要给出数据增强后的效果图，经过实验对比，数据增强后的目标检测性能得到了显著的提升。

图 8. 通过 Mosaic 数据增强后，4 张图片随机裁剪并形成一张新的图片

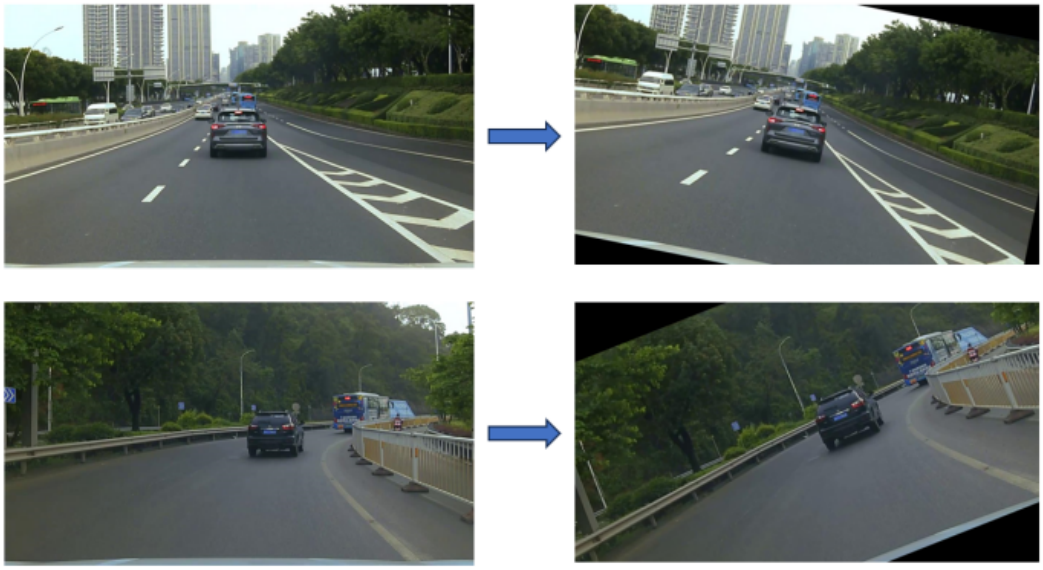

图 9. 通过随机扰动数据增强后，训练样本旋转、平移形成新的训练数据

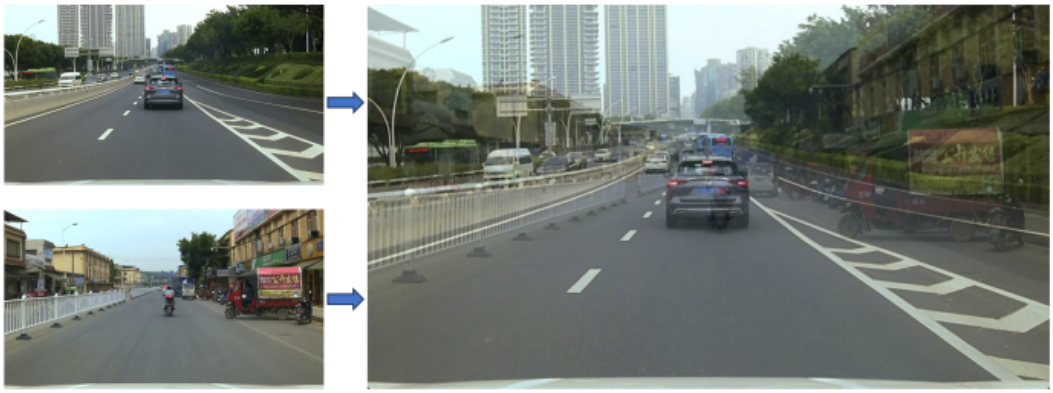

图 10. 通过 Mixup 数据增强混合训练样本，形成更丰富多样的训练数据

图 11. 通过颜色扰动数据增强产生新的训练样本数据

4.4 创新点

由于数据集中实例目标尺寸的特异性（许多实例的尺寸很小），所以设计的模型需要根据特定的任务来进行优化改进。学生发现，输入模型的分辨率对于模型的表现性能具有一定的影响。根据实验结果得出，提高输入模型的图像分辨率，目标检测的性能有一定提升。假设原图的尺寸是 1280×1280，通过预处理阶段将输入图像等比例缩放为 640×640，那么经过 backbone 特征提取后，输入检测头的特征图假设为为 20×20、40×40、80×80。显然，小目标在 80×80 的特征图上的检测效果更佳。(特征图仅仅比原图缩减了 8 倍)。但是这一过程中存在一定的信息丢失。在 1280×1280 缩放为 640×640 的过程当作，虽然图像的尺度信息是等比例缩放的，但是像素值的减半会导致预处理后的图像的细节信息丢失。例如待检测目标的轮廓边缘信息，这对于小目标的影响更加严重。受到该启发，模型在进行特征图融合拼接的过程中，对于 20×20 采样到 40×40，再与 40×40 进行拼接的过程中，采样得到的信息也势必会存在缺陷和不足。为此，我们提出了一个 FGConcat 模块。它对于两个特征图并非简单地进行 Concat 拼接融合，而是先通过 1×1 的卷积对它们的特征进行通道统一，然后使用 EMA 注意力机制 [6] 来更好的融合浅层语义信息和底层语义信息，实现更细粒度的特征融合。考虑到人机目标检测任务对实时性的要求，在所设计的 FGConcat 中 EMA 注意力机制对于输入的两个特征权重共享，这能降低所设计模块的参数量和计算量，在精度和速度之间做出有效权衡。

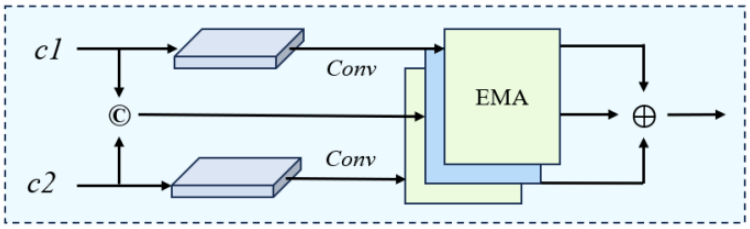

图 12. FGConcat 模块图

所改进的方法的流程图如图 13所示：

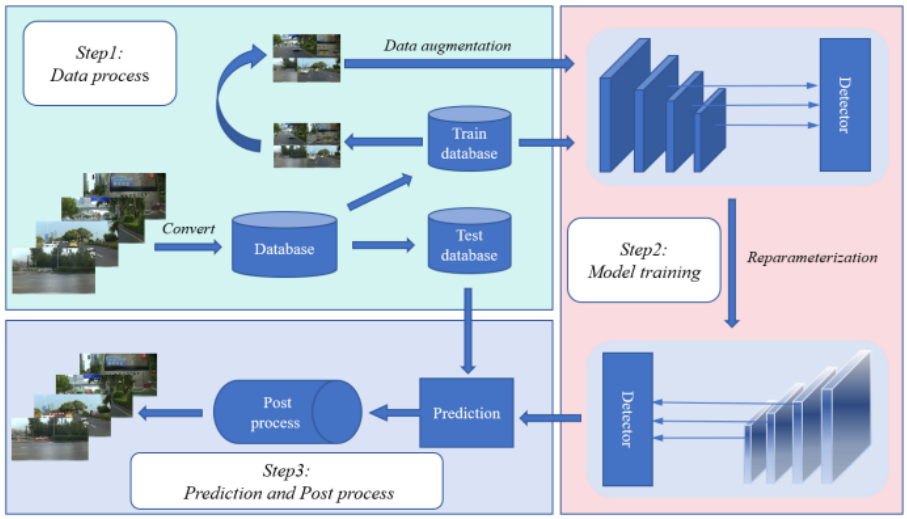

图 13. 方法流程图

5 实验结果分析

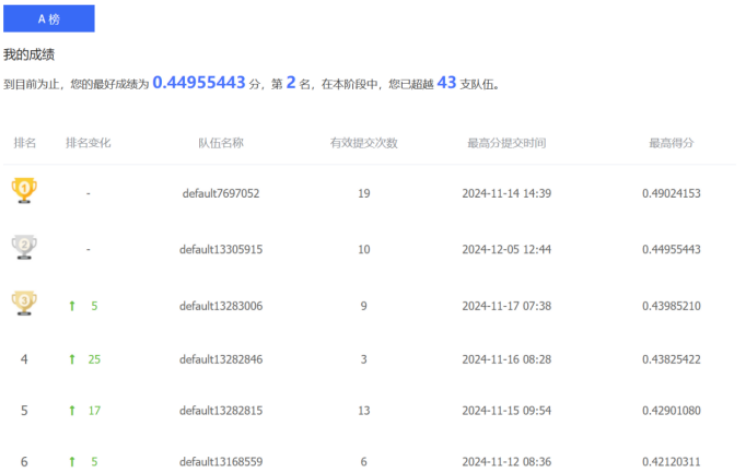

图 14. 竞赛结果排行榜结果

所改进的模型在赛方所提供的排行榜上达到了第二名的成绩。该排行榜的前几名也都是 2024 年 11 月左右提交的，所以也能够侧面反应出本文所改进的模型具有一定的性能提升。但是相比于第一名而言，检测的指标还相对较低，不知其是否使用的是实时的目标检测模型。下表是不同分辨率图片输入模型得到的 mAP 指标分数，可以得出结论，没有经过缩放处理的图像分辨率 1280 的目标检测性能显著优于分辨率为 640 的实验结果。导致这一结果的原因很大可能是由于缩放图像导致待检测目标的边缘轮廓信息丢失，导致目标检测模型没有足够的检测能力来检测出目标；再加上数据集中实例都比较小（上文中可视化过数据集），缩放图片会进一步加剧小目标的细节丢失问题。为了更直观的表示人车目标检测模型的效果，下图展示了在 test_images 中进行预测的结果。test_images 是赛方提供的用来评价模型性能的数据集，本身没有提供坐标标签。由可视化结果可知，所改进的目标检测模型对于小目标的检测效果表现的更好，能够更细粒度地检测出小目标。

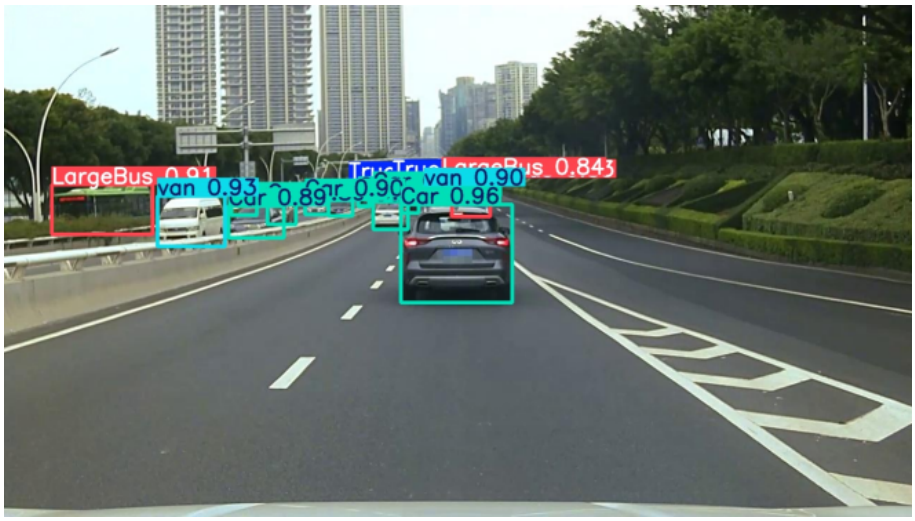

图 15. 人车检测效果图 1

表 1. 实验结果图

Model	Imgsz	mAP
YOLOv8	640	0.415
YOLO11		0.418
RT-DETR		0.416
Ours		0.423
YOLOv8	1280	0.427
YOLO11		0.435
RT-DETR		0.433
Ours		0.449

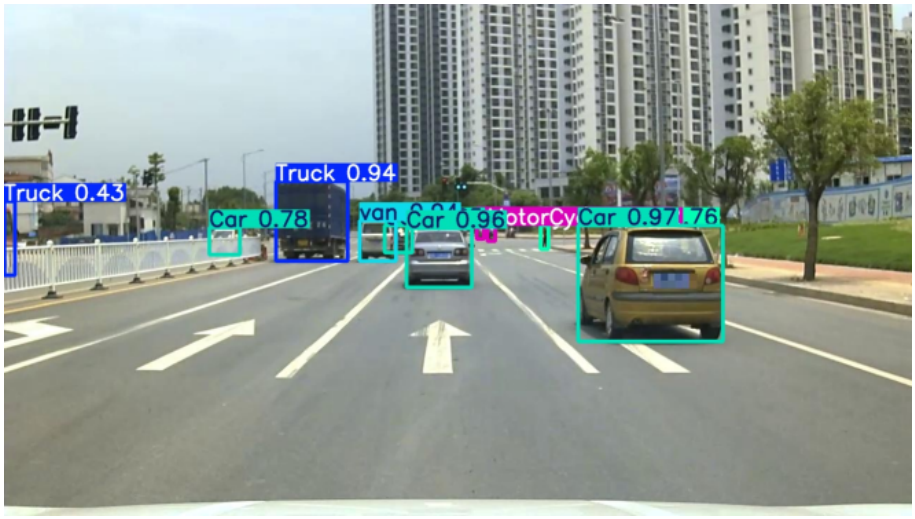

图 16. 人车检测效果图 2

图 17. 人车检测效果图 3

6 总结与展望

本部分对整个文档的内容进行归纳并分析目前实现过程中的不足以及未来可进一步进行研究的方向。RT-DETR 是一个最新的实时端到端的基于 Transformer 的目标检测框架。为了验证其性能，学生在人车目标检测系统上进行相关的实验。人机目标检测是一项重要的人工智能应用项目，这个任务对于未来实现汽车智能驾驶具有一定的价值。在考虑检测目标的召回率以及准确性时，同时也需要重点关注目标检测的实时性，因为只有达到了所需求的实时性，该技术才有可能真正应用到实践当中。考虑到这一隐式的任务需求，学生选取了目前最主流的两个实时目标检测框架进行对比实验，它们分别是 YOLO 家族，以及 RT-DETR。在数据可视化以及预处理阶段，由数据集本身的特点可知，人机类别的种类多（22 种），但是存在类别不均衡的问题，这对于少样本的类别的检出率产生了巨大的挑战。以 YOLO 以及 RT-DETR 作为基线模型来测试该数据集，取得了良好的性能表现。在实践过程中，发现输入模型的图像分辨率对于最终目标检测的性能有所影响，受到启发，提出了一种新颖的特征融合方式：FGConcat。所设计的 FGConcat 可以作为一个即插即用的组件来平替 Concat 操作。实验结果表明，与直接的特征图拼接相比，FGConcat 能够有效提升模型的性能。

后续可以使用 GAN 等模型来生成少样本的数据，来权衡实例的类别分布。针对数据集种小目标的检测问题，可以额外设置一个检测头，作用于更大尺度的特征图上，来提升小目标物体的检出率，但这会带来些许检测速度的下降。

参考文献

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[2] Qiang Chen, Xiaokang Chen, Gang Zeng, and Jingdong Wang. Group detr: Fast training convergence with decoupled one-to-many label assignment. *arXiv preprint arXiv:2207.13085*, 2(3):12, 2022.

[3] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13619–13627, 2022.

[4] Junyu Lin, Xiaofeng Mao, Yuefeng Chen, Lei Xu, Yuan He, and Hui Xue. D^ 2etr: Decoder-only detr with computationally efficient cross-scale attention. *arXiv preprint arXiv:2203.00860*, 2022.

[5] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv preprint arXiv:2201.12329*, 2022.

[6] Daliang Ouyang, Su He, Guozhong Zhang, Mingzhu Luo, Huaiyong Guo, Jian Zhan, and Zhijie Huang. Efficient multi-scale attention module with cross-spatial learning. In *ICASSP*

2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1–5. IEEE, 2023.

[7] Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16965–16974, 2024.

[8] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.