[1] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End Object Detection with Transformers," p. arXiv:2005.12872. [Online]. Available: https://ui.adsabs.harvard.edu/abs/2020arXiv200512872C

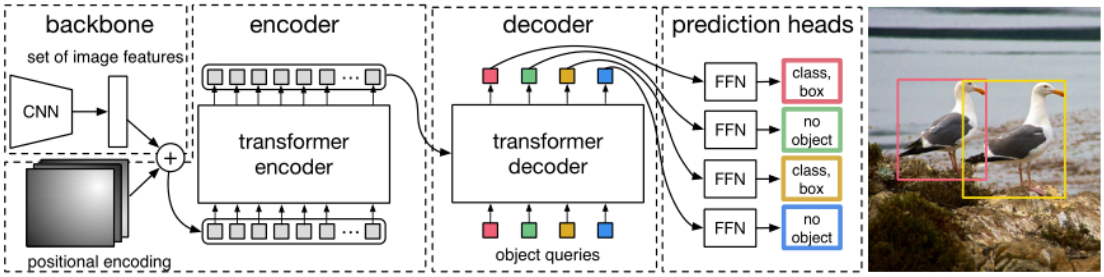
- 主题：利用 Transformer 的端到端目标检测
- 解决问题：
  目标检测：对象→边界框&类别
  现代目标检测器的性能受到影响：重复预测后处理、锚集设计和边界框分配的启发式→端到端的直接集预测
- 核心内容：一种基于直接集预测和 transformer 的目标检测新方法——检测转换器（DETR）
  DETR 根据目标间关系和全局图像的语境，直接并行输出最终预测集。
  DETR 的特点：双向匹配损失+并行解码 transformer
  Transformer 的编码器-解码器架构
  Transformer 的 self-attention 机制：明确了序列中元素之间的所有成对交互，适合集合预测的约束
  一次预测所有对象，使用集合损失函数（预测值和真实值的双向匹配）进行端到端训练，无需自定义层而可轻松复制。
- 概念：
  ① 集合预测的双相匹配：
  集合预测任务是多标签分类，基线方法（一对一）不适合检测元素间存在底层结构
  避免重复，现代预测使用后处理，但直接集预测无需后处理
  直接集预测：全局推理的目标关系建模结构，利用匈牙利算法设计损失函数以强制排列不变性
  ② 基于 transformer 的编码器-解码器架构，并行解码：
  Attention 机制：从整个序列中聚合信息的神经网络层
  Transformer 中 self-attention 层，扫描序列元素、聚合序列信息以更新
  优点：全局计算能力和记忆能力，比 RNN 更适合长序列
  ③ 目标检测：
  现代目标检测：基于初始猜测进行预测，两阶段检测器用建议框，单阶段检测器用锚或网格
  基于集合的损失：用 attention 对不同预测之间的关系进行建模，早期模型仅用 Cov、FC 层建模，需手工设计 NMS
  递归检测器：基于 CNN 激活的编码器-解码器架构的双向匹配损失直接生成一组边界框，小数据集评估，基于自回归模型（RNN）
- 模型：DETR 模型
  原理：目标检测集合预测损失
  <1> 寻找最小损失时的排列组合
  <2> 计算全局损失函数（平衡损失项）
  目标和∅之间的损失为常数，用概率表示类别预测损失以使两种损失可公度

$$\hat{\sigma} = \underset{\sigma \in \mathfrak{S}_N}{\arg\min} \sum_i^N \mathcal{L}_{match}(y_i, \hat{y}_{\sigma(i)}) \qquad y_i = (c_i, b_i)$$

$$\mathcal{L}_{match} = -\mathbb{1}_{\{c_i \neq \varnothing\}} \hat{p}_{\sigma(i)}(c_i) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{box}(b_i, \hat{b}_{\sigma(i)})$$

$$\mathcal{L}_{box}(b_i, \hat{b}_{\sigma(i)}) = \lambda_{iou} \mathcal{L}_{iou}(b_i, \hat{b}_{\sigma(i)}) + \lambda_{L1} \| b_i - \hat{b}_{\sigma(i)} \|_1$$

架构：DETR=主干网络（CNN）+编码器-解码器 transformer+检测预测的 FFN



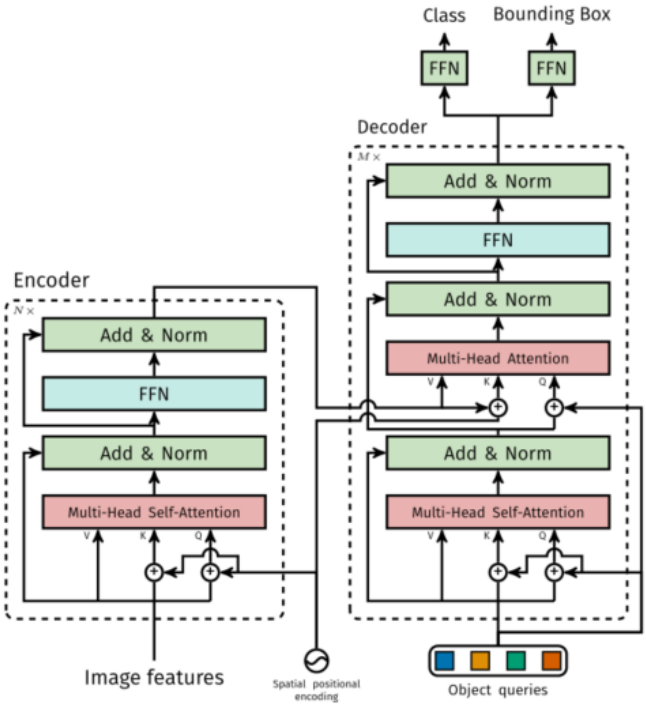
① 主干网络（提取袖珍特征）：低分辨率激活图（2048, 1/32, 1/32）

② Transformer 编码器：一维序列输入，每层=多头 self-attention 模块+FFN，固定的空间位置编码添加到了每层输入

③ Transformer 解码器：多头编码器-解码器 self-attention 机制，输入嵌入（学习的位置编码，d, N），目标查询转换为输出嵌入，全局推理

④ 检测预测网络：简单前馈网络（FFN），ReLU 激活函数，三个隐藏层，每层维度为 d，根据输入图像预测归一化中心坐标、高、宽，所有 FFN 共享参数

⑤ 附加损失：每个解码器层后添加预测 FFN 和匈牙利损耗



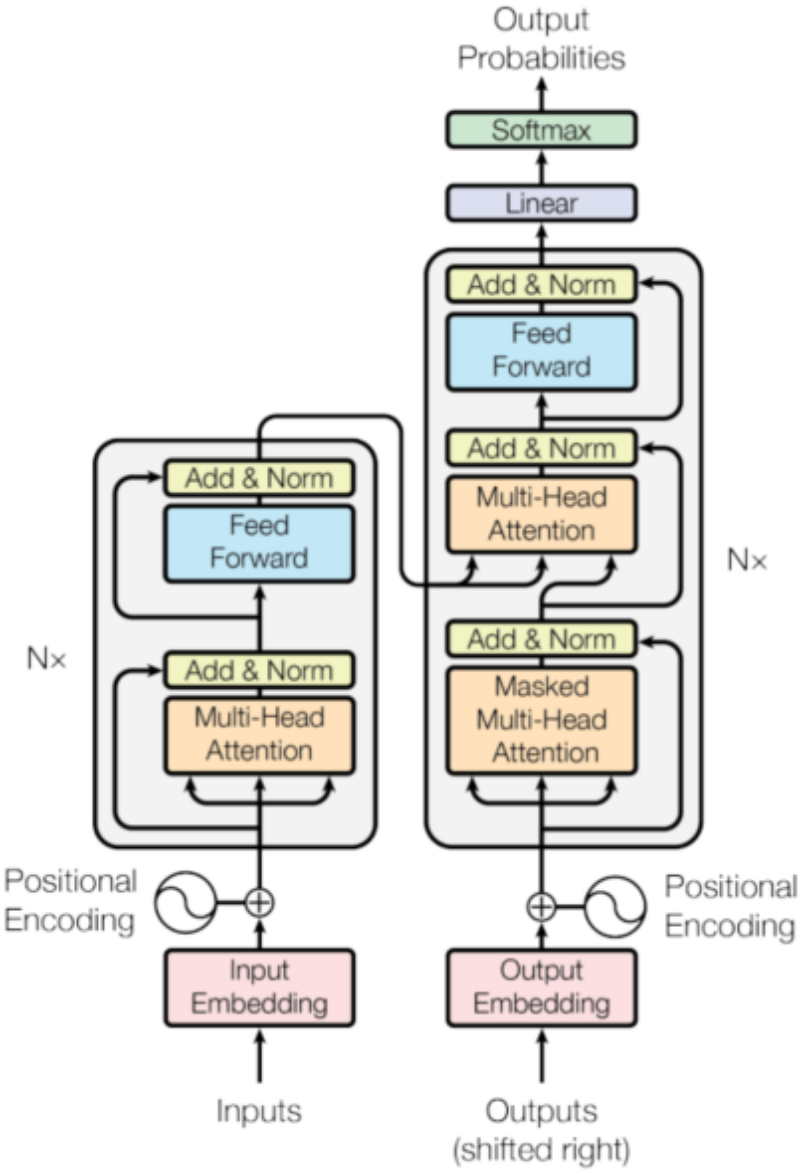
- 数据集：目标检测数据集 COCO2017、全景分割数据集
- 实验：DETR 用 AdamW 训练，transformer 权重用 Xavier 初始化
  DETR 和 DETR-R101：不同主干网络 ResNet50 和 ResNet101
  DETR-DC5 和 DETR-DC5-R101：主干最后加膨胀，首个卷积操作中步长减 1，提高特征分辨率（2 倍）

Model	GFLOPS/FPS	#params	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster RCNN-DC5	320/16	166M	39.0	60.5	42.3	21.4	43.5	52.5
Faster RCNN-FPN	180/26	42M	40.2	61.0	43.8	24.2	43.5	52.0
Faster RCNN-R101-FPN	246/20	60M	42.0	62.5	45.9	25.2	45.6	54.6
Faster RCNN-DC5+	320/16	166M	41.1	61.4	44.3	22.9	45.9	55.0
Faster RCNN-FPN+	180/26	42M	42.0	62.1	45.5	26.6	45.4	53.4
Faster RCNN-R101-FPN+	246/20	60M	44.0	63.9	**47.8**	**27.2**	48.1	56.0
DETR	86/28	41M	42.0	62.4	44.2	20.5	45.8	61.1
DETR-DC5	187/12	41M	43.3	63.1	45.9	22.5	47.3	61.1
DETR-R101	152/20	60M	43.5	63.8	46.4	21.9	48.0	61.8
DETR-DC5-R101	253/10	60M	**44.9**	**64.7**	47.7	23.7	**49.5**	**62.3**

- 性能比较：DETR 和 R-CNN，DETR 的性能在大目标上更好，小目标上低（self-attention）
- 剥蚀分析：架构的组件如何影响性能
  ResNet-50 的 DETR 模型，6 编码器层 6 解码器层
  编码器层数：编码器层数↑，AP↑
  解码器层数：解码器层数↑，AP↑，且在层数>2 时有无 NMS 影响不大
  FFN：等效于 1×1 卷积层，在移除后 AP 显著↓
  空间位置编码：移除后 AP↓
- 效果：简化了目标检测流程，消除了手工设计组件的需求（NMS、锚生成），可以在任何框架中实现和重复使用

[2] A. Vaswani *et al.*, "Attention Is All You Need," p. arXiv:1706.03762. [Online]. Available: https://ui.adsabs.harvard.edu/abs/2017arXiv170603762V

- 解决问题：时序模型（RNN、LSTM、GRU 等），语言模型、编码器-解码器模型，时间上并行度低，早期时序信息在后期容易丢失而导致存储要求高
  Attention
- 核心内容：transformer，将递归层换成多头 self-attention
  纯基于 attention 机制，并行度高，短时间内达到了当前最优
  可以同时观察到所有像素信息，多头模拟 CNN 多输出通道机制
  Self-attention 机制
- 模型：transformer
  编码器-解码器结构：编码同时输入 x（n），生成码向量 z（n），逐个解码输出 y（m），自回归（过去的输入是模型的输出）
  将 self-attention 层和逐点全连接层堆叠在一起



- 实验：两个机器翻译，8 个 GPU 上 3.5 天