

DETR

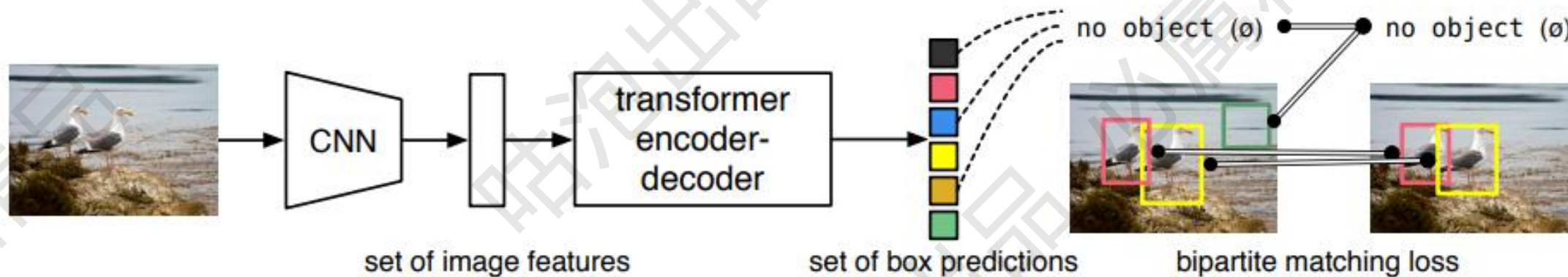
- ✓ 说到目标检测你能想到什么
 - ✎ faster-rcnn系列，开山之作，各种proposal方法
 - ✎ YOLO肯定也少不了，都是基于anchor这路子玩的
 - ✎ NMS那也一定得用上，输出结果肯定要过滤一下的
 - ✎ 如果一个目标检测算法，上面这三点都木有，你说神不神！

DETR

✓ 基本思想

✎ 先来个CNN得到各Patch作为输入，再套transformer做编码和解码

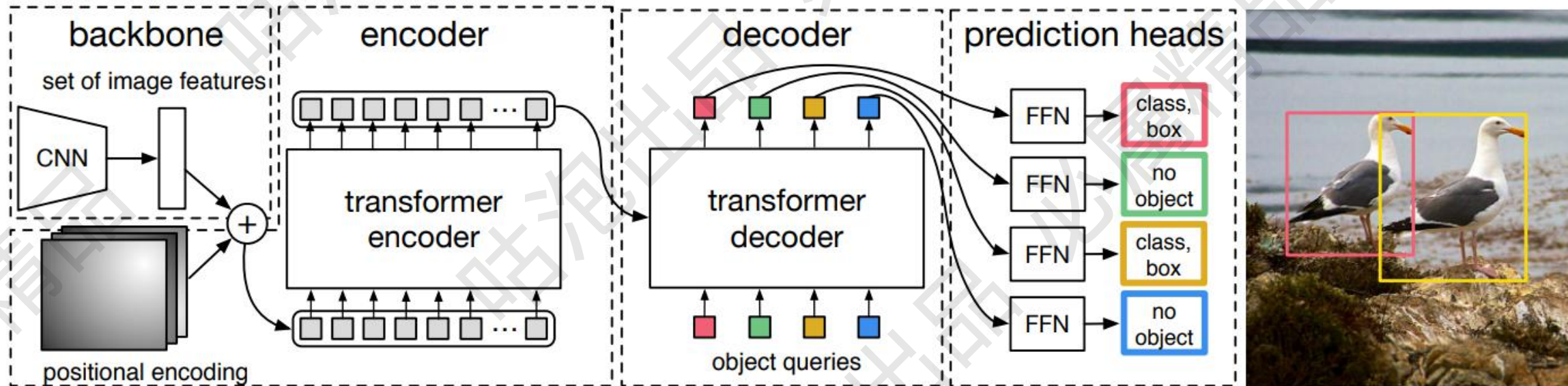
✎ 编码路子跟VIT基本一样，重在在解码，直接预测100个坐标框



DETR

✓ 整体网络架构

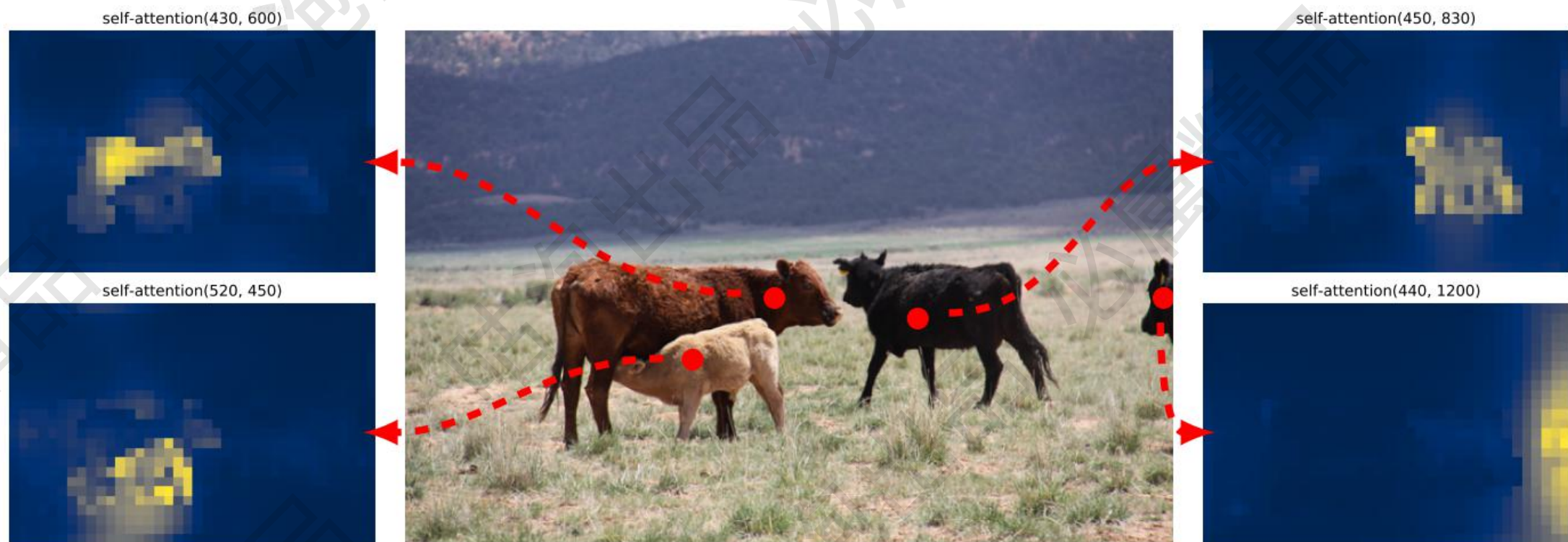
📌 object queries是核心，让它学会怎么从原始特征找到是物体的位置



DETR

✓ Encoder完成的任务

📎 得到各个目标的注意力结果，准备好特征，等解码器来选秀



DETR

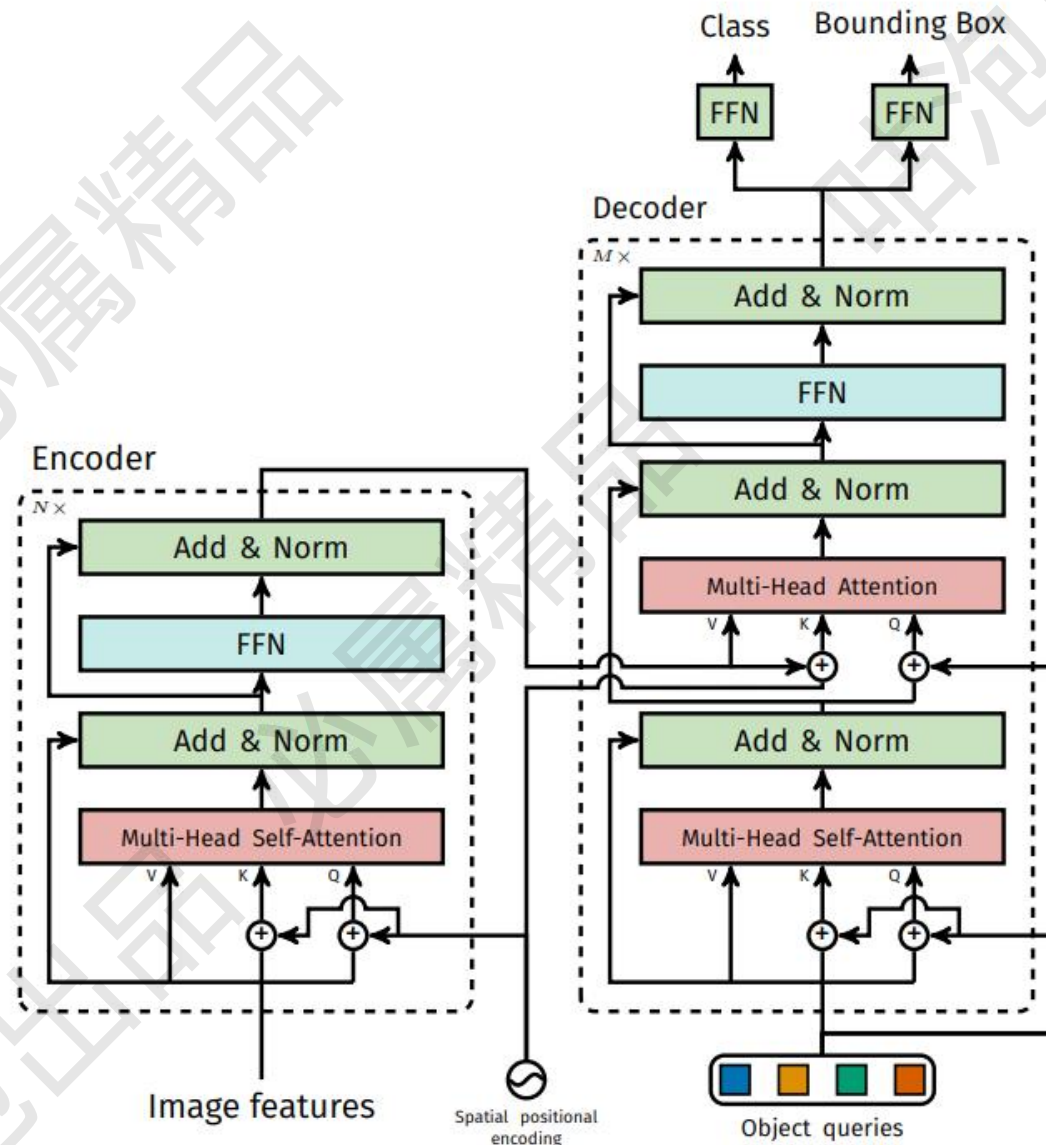
✓ 网络架构

✎ 输出层就是100个object queries预测

✎ 编码器木有啥特别的，正常整就行

✎ 解码器首先随机初始化object queries
(0+位置编码，简直惊呆。。。)

✎ 通过多层让其学习如何利用输入特征

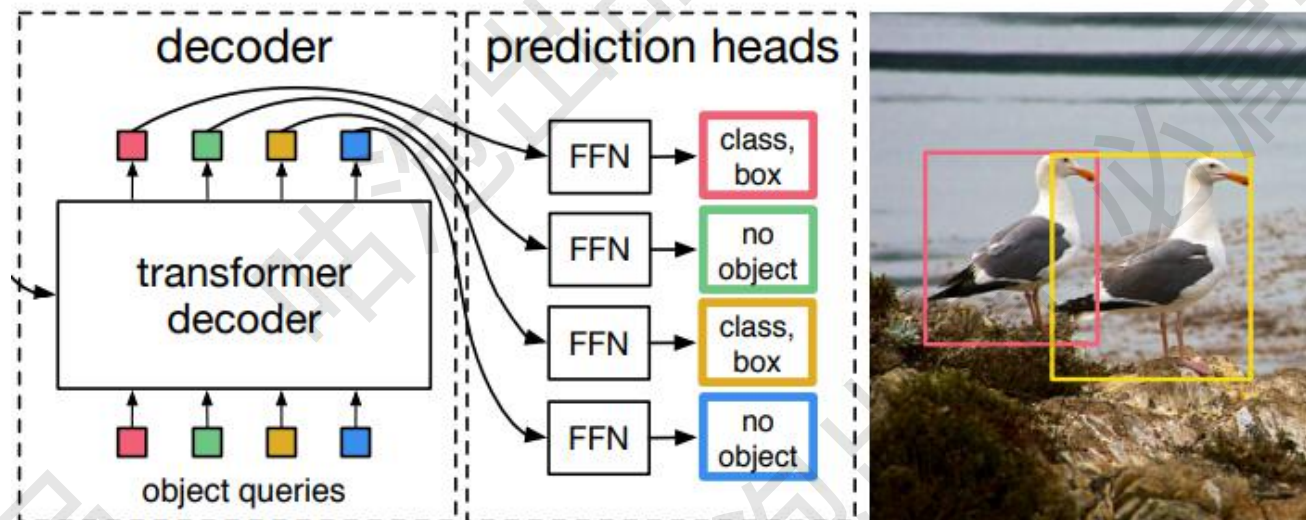


DETR

✓ 输出的匹配

✎ GT只有两个，但是预测的恒为100个，怎么匹配呢？

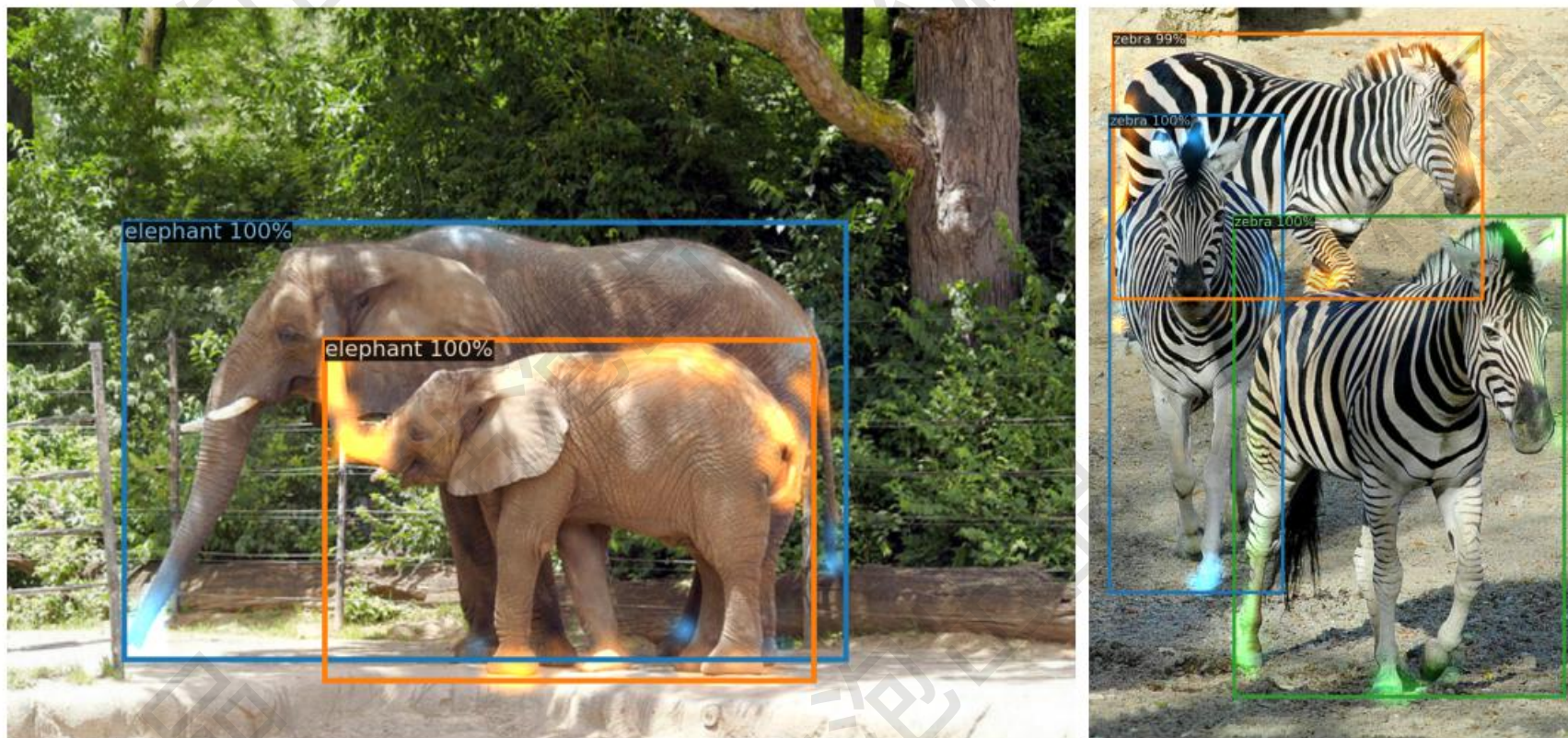
✎ 匈牙利匹配完成，按照LOSS最小的组合，剩下98个都是背景



DETR

✓ 注意力起到的作用

📌 这个注意力挺有意思，能不被遮挡，照样可以学出来（注意颜色）



DETR

✓ 小的细节

✎ decoder中的位置肯定最重要了，这个得学习才行；每层都预测（Auxiliary）

✎ 100个预测框之间可以相互通信（这是我地，你去那边瞅瞅吧）

✎ 训练用了多个卡，感觉家里条件一般的够呛能整个起

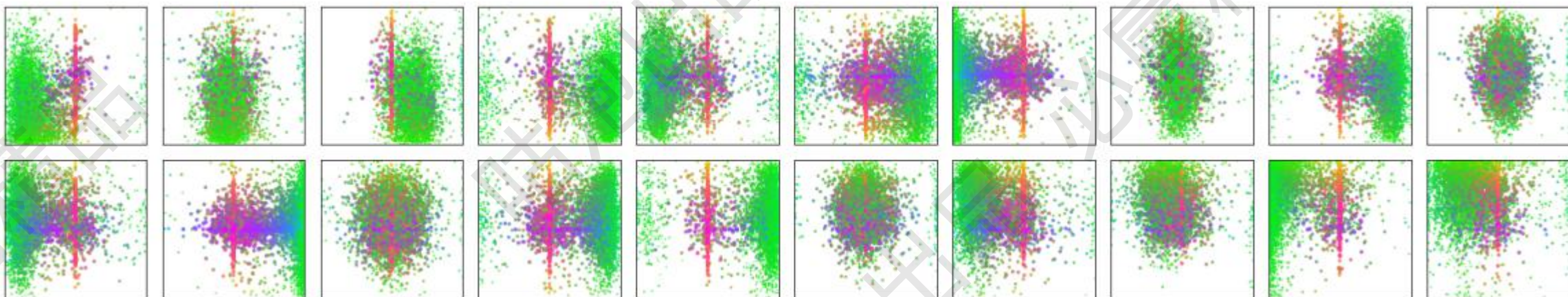
ing images once. Training the baseline model for 300 epochs on 16 V100 GPUs takes 3 days, with 4 images per GPU (hence a total batch size of 64). For the longer schedule used to compare with Faster R-CNN we train for 500 epochs with learning rate drop after 400 epochs. This schedule adds 1.5 AP compared to the shorter schedule.

DETR

✓ 100个兄弟各自要干啥

✎ 论文中可视化了其中20个，绿色是小物体，红蓝是大物体

✎ 基本描述了各个位置都需要关注，而且它们还是各不相同的



DETR

✓ 额外证明

📎 transformer不仅在检测领域好使，分割里照样行
(感觉就像是让一群人去做分割，每个人做其中一块，最后合并一起)

