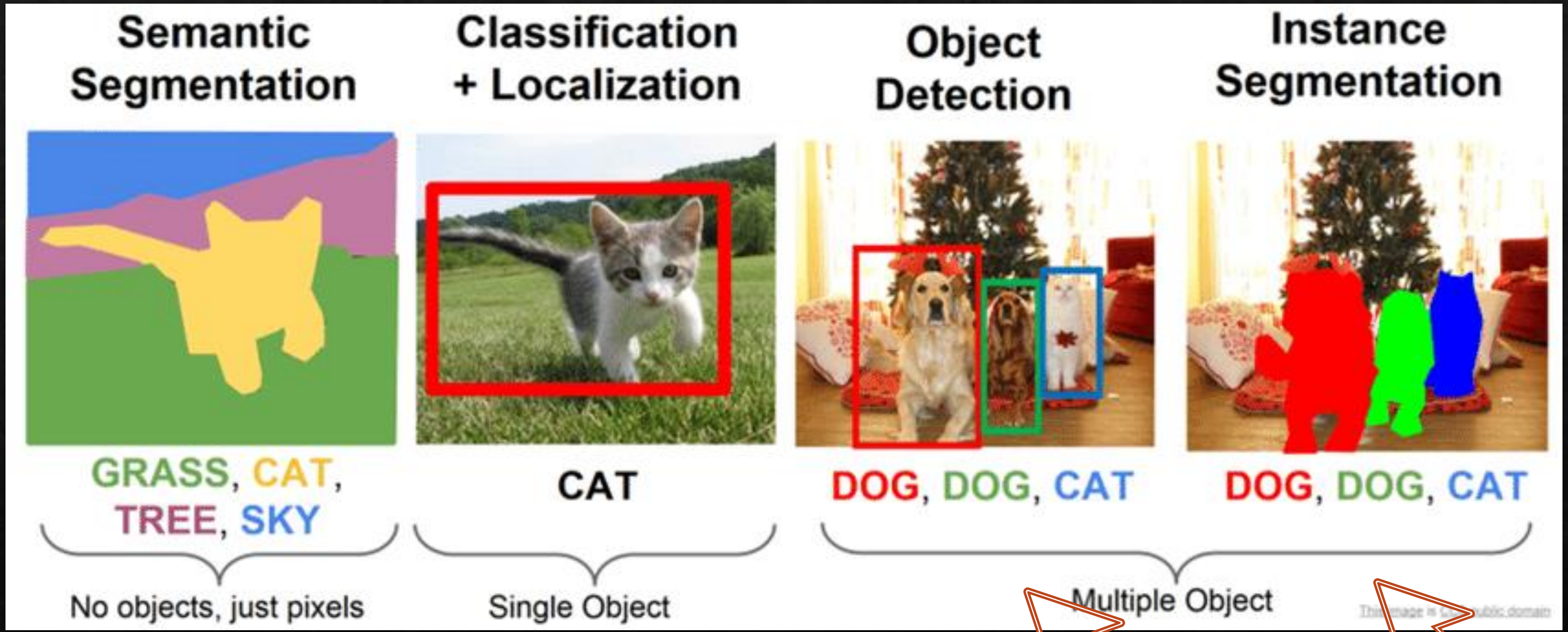
VJAI 2020
By Le Anh

End-to-End Object Detection with Transformers (DETR)

DETR



DETR DETR

Agenda

- DETR
  - Hungarian Algorithm
    - Weighted bipartite graph
  - DETR architecture
  - Encoder-Decoder
  - FPN
  - Loss Function
  - Object Query
- Summary

Introduction to DETR

DETR stands for End-to-End Object Detection with Transformers

New method that views object detection as a direct set prediction problem

- Output is fixed set of N objects

Removing the need for many hand-designed components

- non-maximum suppression
- anchor generation

Compare to Faster RCNN

- significantly better performance on large objects
- lower performances on small objects

Term

- Bounding Box
- ROI
- IOU
- Transformer
    - Attention
- Fully Connection Network (FCN)

Hungarian Algorithm

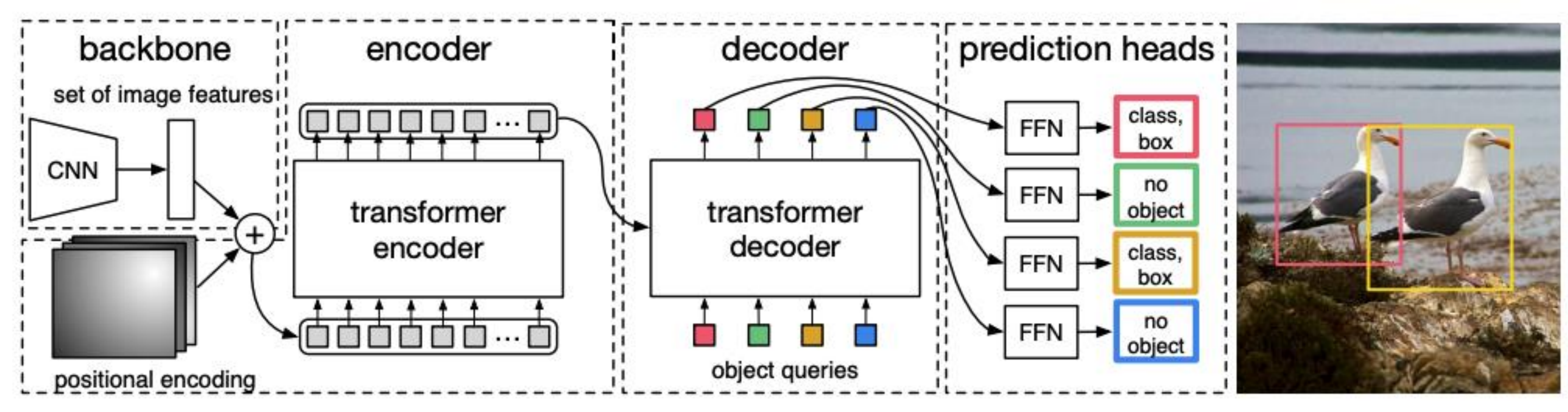
- Problems
  - A factory needs to assign each of 4 workers to one of 4 tasks
  - Aim: Determine the best assignment of workers to tasks so that the overall time taken is minimized

$$
\begin{array}{l c}
 & \begin{array}{cccc} A & B & C & D \end{array} \\
\begin{array}{l} \text{Wendy} \\ \text{Xenefon} \\ \text{Yolanda} \\ \text{Zelda} \end{array} & \begin{bmatrix} 30 & 40 & 50 & 60 \\ 70 & 30 & 40 & 70 \\ 60 & 50 & 60 & 30 \\ 20 & 80 & 50 & 70 \end{bmatrix}
\end{array}
$$

Result
Zelda – A : 20 mins
Yolanda – D : 30 mins
Wendy – B or C : 40 min or 50 min
Xenefor - C or B : 50min or 40min

Overall: 140mins

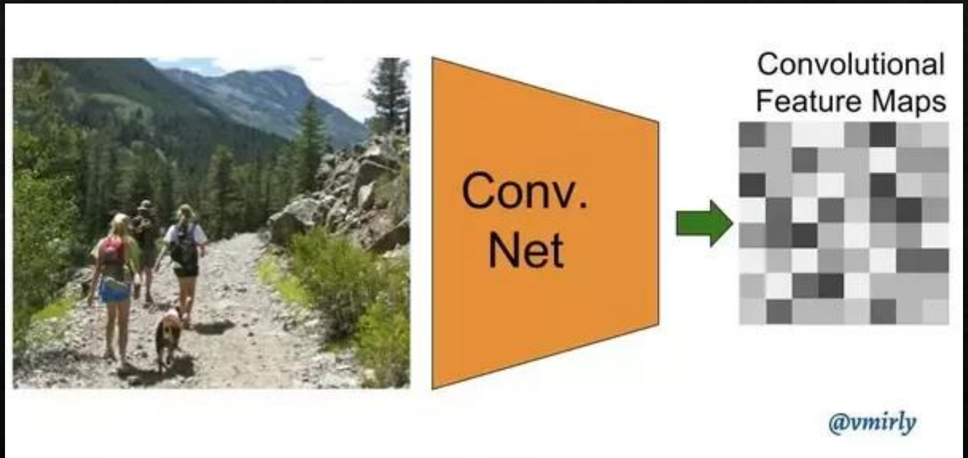
DETR architecture



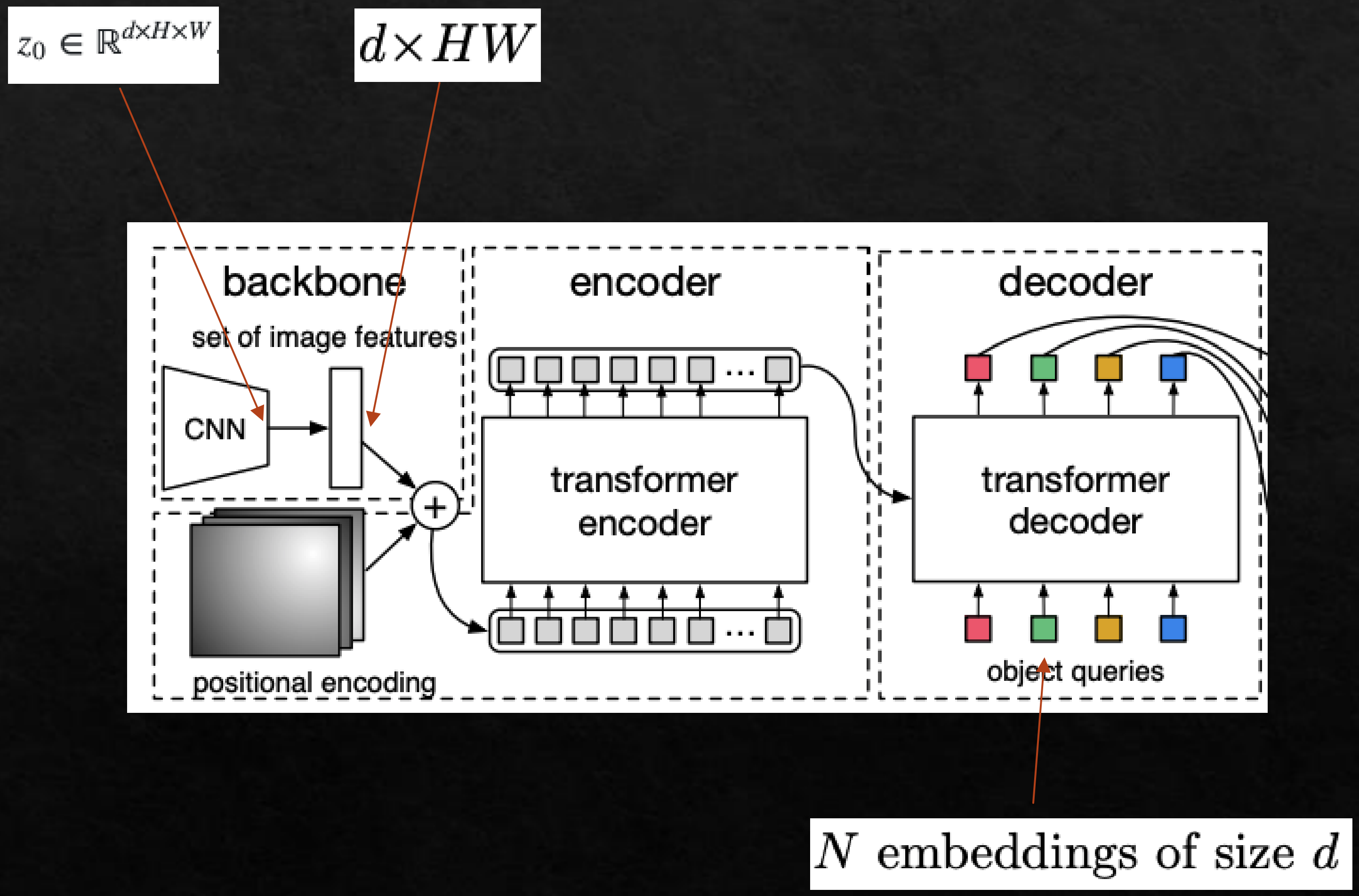
DETR architecture

$x_{\text{img}} \in \mathbb{R}^{3 \times H_0 \times W_0}$

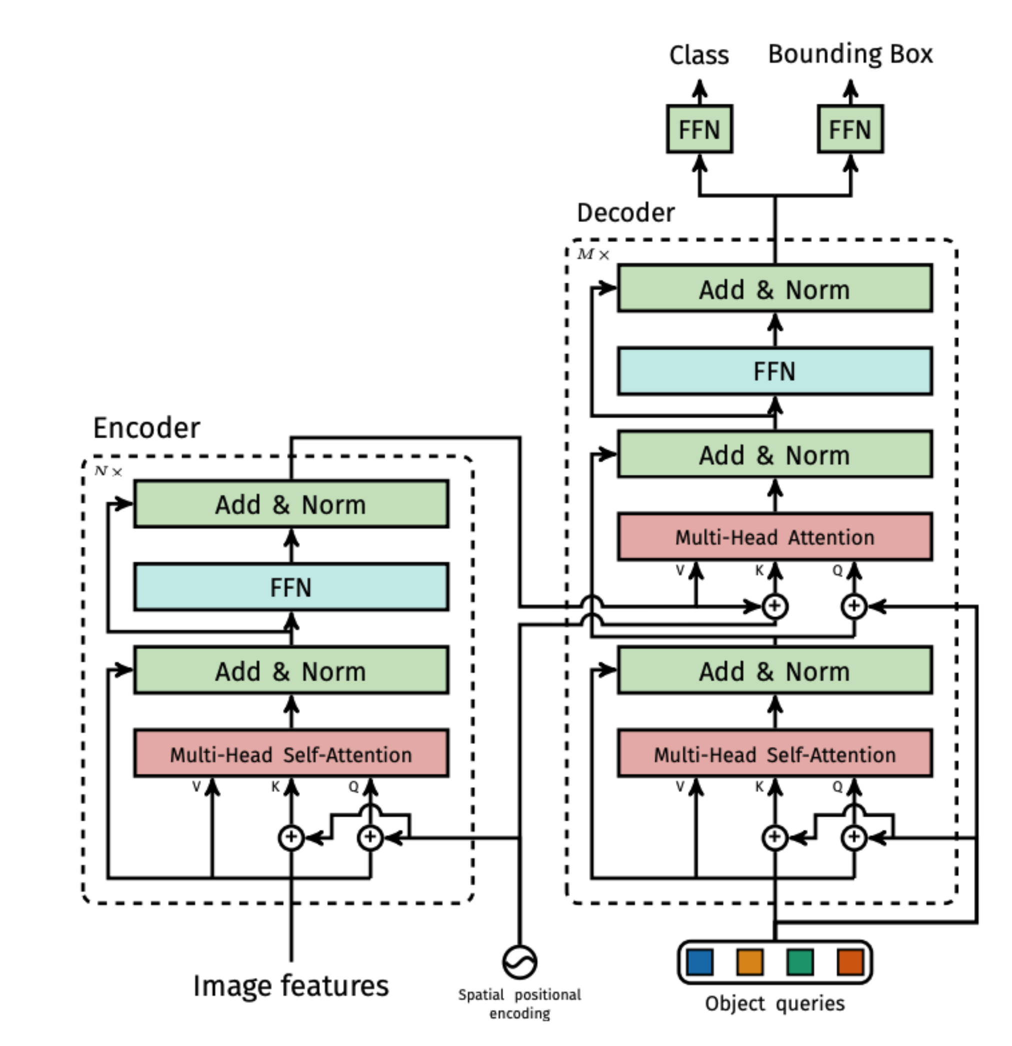
$f \in \mathbb{R}^{C \times H \times W}$



Transformer

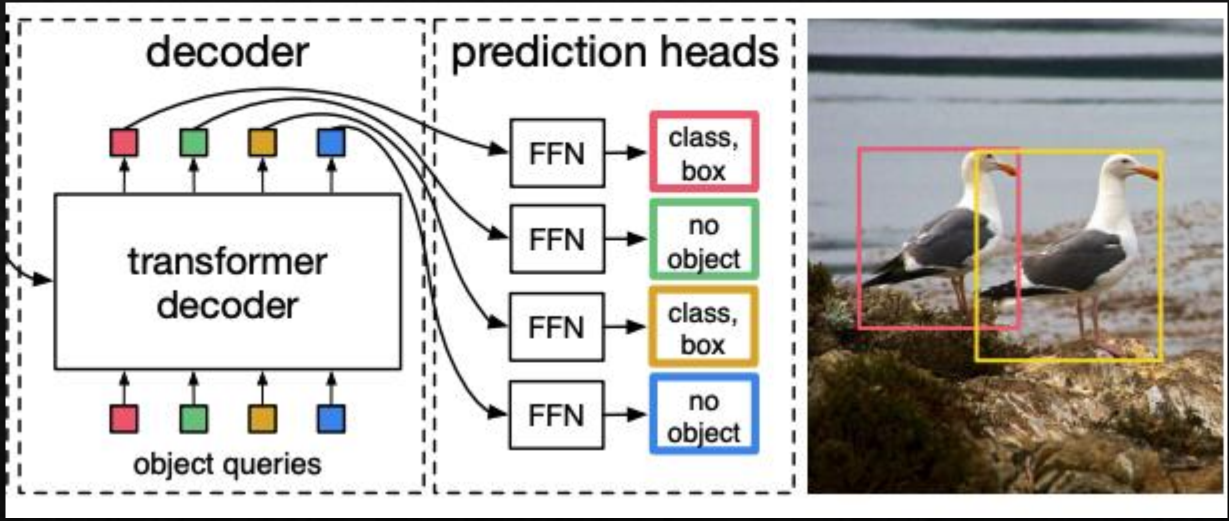


Transformer

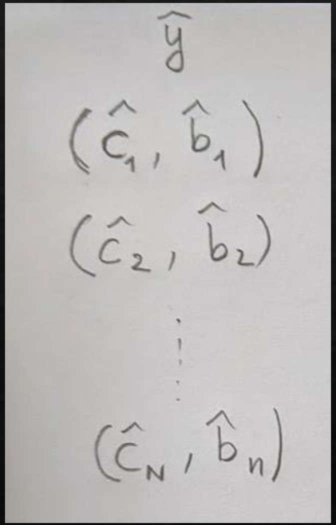


FCN

```
self.class_embed = nn.Linear(hidden_dim, num_classes + 1)
self.bbox_embed = MLP(hidden_dim, hidden_dim, 4, 3)
```



Output



Loss Function

Step 1: find a bipartite matching between these two sets
Step 2: compute the loss function, the Hungarian loss for all pairs matched in the previous step

Loss Function

Step 1: find a bipartite matching between these two sets

- Pair matching cost between ground truths and predictions

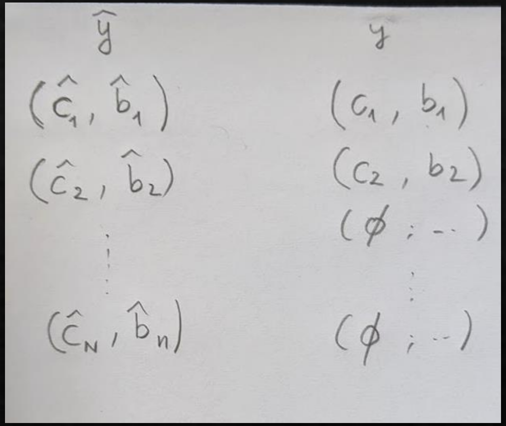
$$-\mathbb{1}_{\{c_i \neq \varnothing\}} \hat{p}_{\sigma(i)}(c_i) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(b_i, \hat{b}_{\sigma(i)})$$

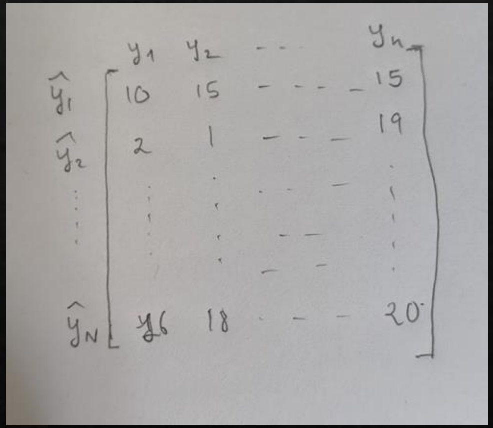
- find one-to-one matching for direct set prediction without duplicates

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

Loss Function

Step 1: find a bipartite matching between these two sets





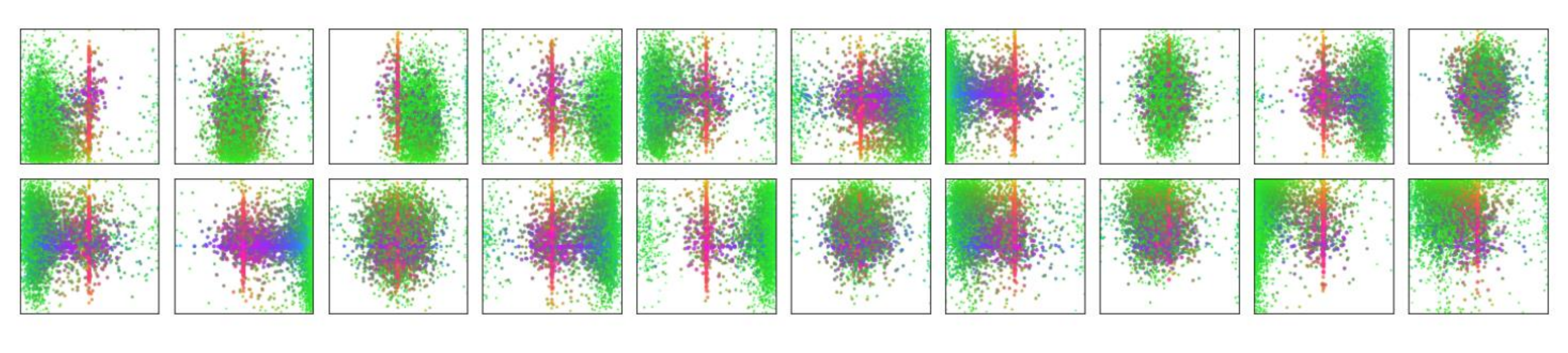
Loss Function

Step 2: compute the loss function, the Hungarian loss for all pairs matched in the previous step

$$\mathcal{L}_{\text{Hungarian}}(y, \hat{y}) = \sum_{i=1}^{N} \left[-\log \hat{p}_{\hat{\sigma}(i)}(c_i) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(b_i, \hat{b}_{\hat{\sigma}}(i)) \right]$$

Object Query

- Trainable parameters
- each slot learns to specialize on certain areas and box sizes with several operating modes



Encoder Self Attention

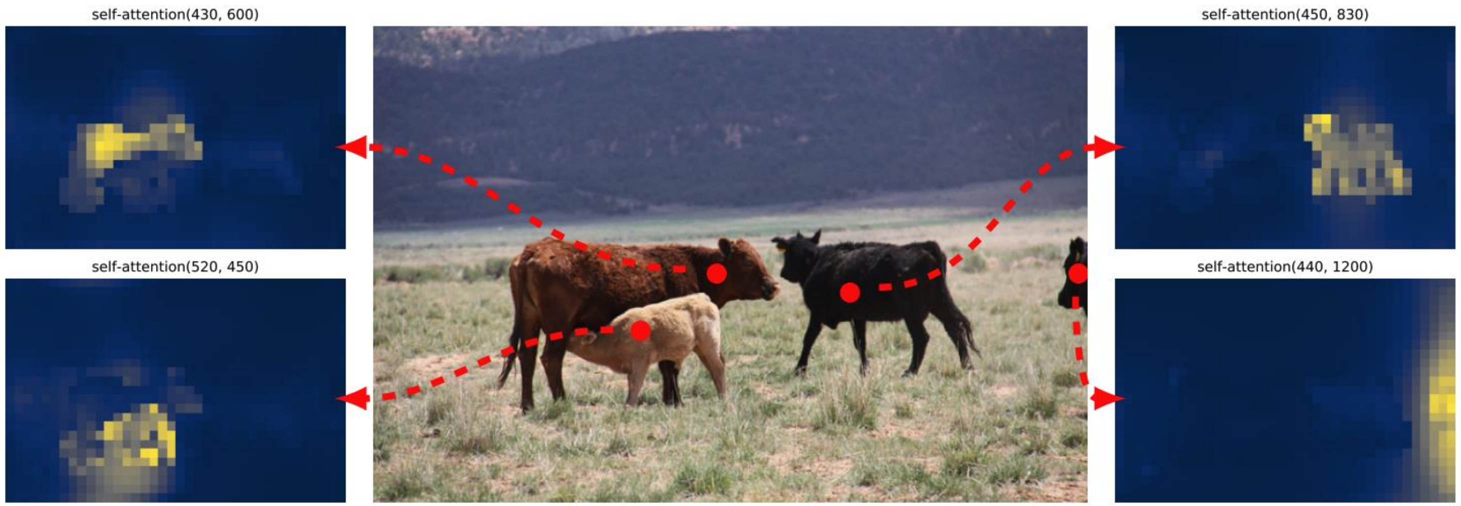


Fig. 3: Encoder self-attention for a set of reference points. The encoder is able to separate individual instances. Predictions are made with baseline DETR model on a validation set image.

Decoder attention

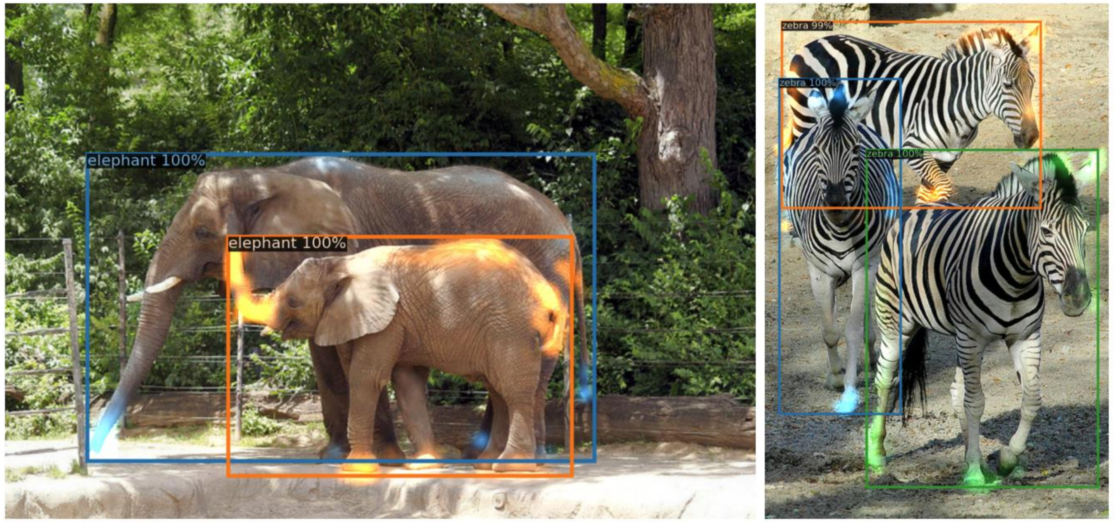


Fig. 6: Visualizing decoder attention for every predicted object (images from COC
`val` set). Predictions are made with DETR-DC5 model. Attention scores are coded wit
different colors for different objects. Decoder typically attends to object extremitie
such as legs and heads. Best viewed in color.

Summary

- Introduction to DETR
- Hungarian algorithms
- Using of Transformer in object detection