End-to-End Object Detection with Transformers

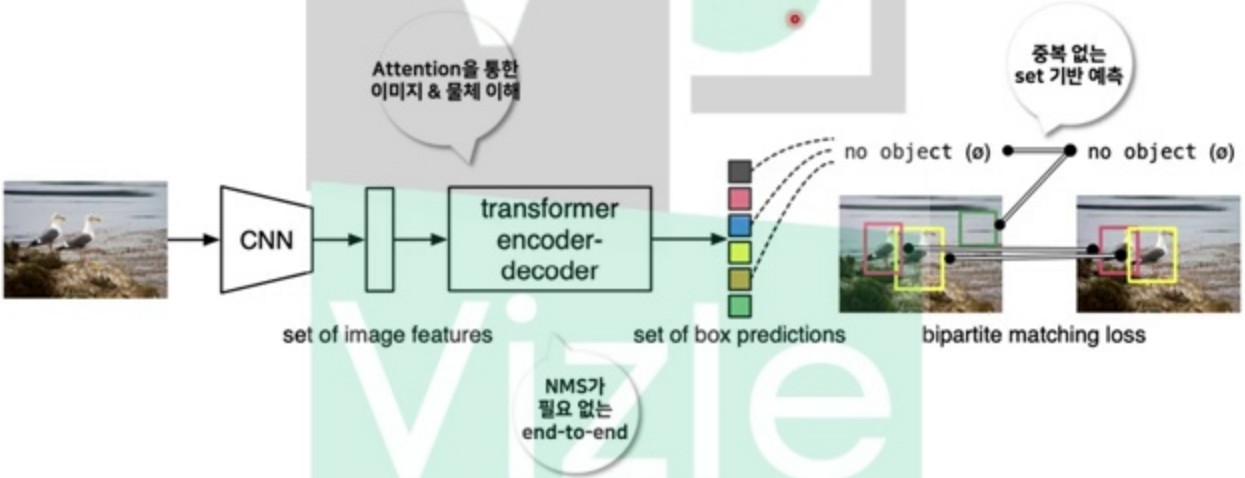
N Carion et al., ECCV, 2020

고려대학교 DSBA연구실
석사과정 소규성

Background

DETR (DEtection with TRansformers)

- ✓ Transformer와 이분 매칭(Bipartite-matching) 기반의 새로운 detection 구조
- ✓ Object detection을 **direct set-prediction**의 문제로 접근하며, end-to-end 모델로서 geometric prior가 필요하지 않음 (RPN, NMS와 같은 hand-crafted 엔지니어링이 필요 없음)
- ✓ 구조적으로 간결함에도 다른 task에 확장성이 높고(e.g. panoptic segmentation), 어텐션 메커니즘에 의해 전역적 정보를 이용함에 따라 큰 물체 탐지에 대해서 Faster R-CNN에 비해 더 높은 성능을 보여줌



Architecture

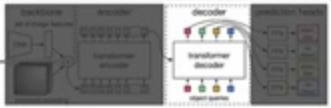
Encoder

- ✓ Encoder는 attention mechanism을 기반으로 feature map의 pixel과 pixel 간의 관계를 학습
- ✓ Locality 중심의 CNN과 다르게 global한 정보를 학습함으로써 이미지를 이해하고,
  특히 object detection task에 맞게 학습됨으로써 이미지 내 object의 위치, 관계 등 또한 학습하게 됨

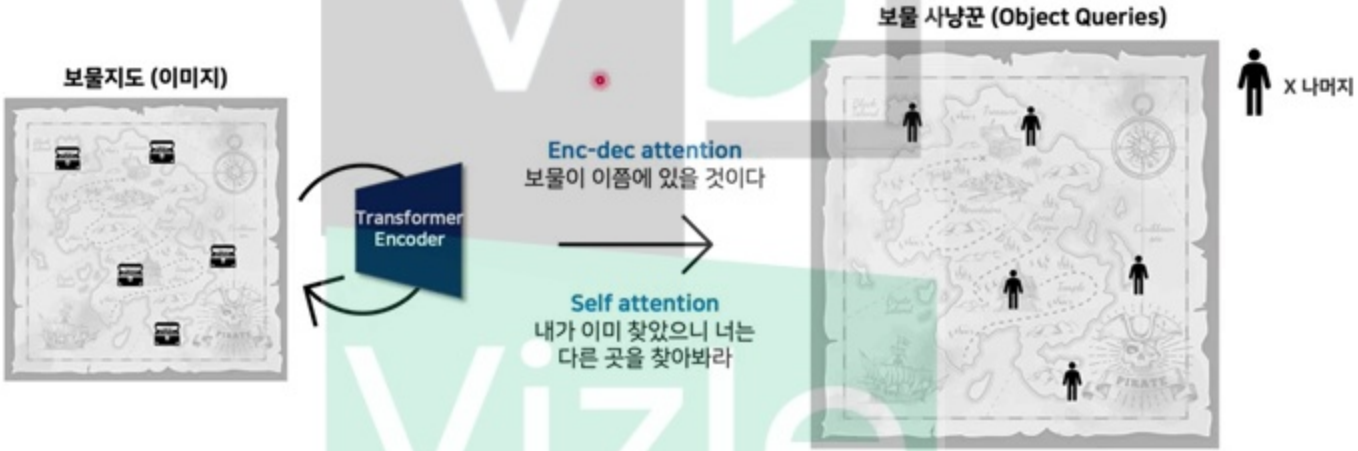
*) https://www.youtube.com/watch?v=utxbUlo9CyY

Architecture

Decoder: Object Queries



- ✓ Object query는 곧 정보를 담기 위한 그릇(slot)으로 생각할 수 있고,
- ✓ Decoder의 ① Encoder-decoder attention을 통해 이미지의 어느 부분을 위주로 봐야할 지 (물체가 어느 위치에 있을 확률이 높을 지),
  ② Self-attention을 통해 자신들의 역할을 어떻게 분배하여 최적의 일대일 매칭을 수행할 수 있을 지를 학습하게 됨



It turns out experimentally that it will tend to reuse a given slot to predict objects in a given area of the image

Training Process

Bipartite Matching (feat. Hungarian Algorithm)

- ✓ Object query마다 예측된 결과물(① Class 유무, ② 박스 좌표)과 ground truth set 간 Hungarian algorithm 기반 매칭 수행
- ✓ 이 때 매칭의 기준으로 pair-wise matching cost인 $\mathcal{L}_{match}$를 활용하며, $\mathcal{L}_{match}$를 최소화하는 최적의 순열($\hat{\sigma}$)를 찾음
- ✓ 이는 Anchor의 비교 방식과 유사하나, anchor는 동일한 ground truth object와의 중복 예측을 허용하는 반면 DETR은 set prediction과 ground truth 간 일대일 매칭 수행하여 중복을 배제

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

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

클래스 예측 Cost — 박스 좌표 예측 Cost

Experiments

DETR vs Faster R-CNN

* -DC5: Dilated Convolution 추가 → Higher Resolution

- ✓ Faster RCNN과의 최대한 합리적인 비교를 위해, 기본적인 Faster RCNN 모델에 아래와 같은 요소를 추가
  - ① Bounding box loss function에 g-IOU Loss를 추가
  - ② DETR 학습 시와 동일한 crop augmentation을 추가
  - ③ 더욱 긴 training schedule(x3)을 이용하여 학습*)
- ✓ 크기가 큰 물체에 대해서는 Faster RCNN 대비 높은 성능을 달성했으나, 크기가 작은 물체에 대해서는 낮은 성능을 기록

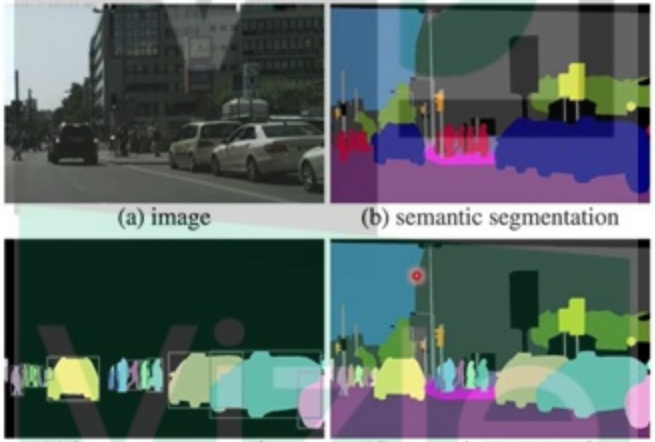
Model	GFLOPS/FPS	#params	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster RCNN-DC5	320/16	166M	39.0	60.5	42.3	21.4	43.5	52.5
Faster RCNN-FPN	180/26	42M	40.2	61.0	43.8	24.2	43.5	52.0
Faster RCNN-R101-FPN	246/20	60M	42.0	62.5	45.9	25.2	45.6	54.6
Faster RCNN-DC5+	320/16	166M	41.1	61.4	44.3	22.9	45.9	55.0
Faster 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**

*) Rethinking ImageNet Pre-training, K. He et al., 2018 (https://arxiv.org/abs/1811.08883)

DETR for Panoptic Segmentation

What is Panoptic Segmentation?

- ✓ Panoptic segmentation은 **① 이미지 내의 모든 픽셀을 사전에 정해진 class로 분류하는 semantic segmentation**과,
  ② 동일한 class 내에서도 서로 다른 객체를 구분하는 instance segmentation을 합친 개념
- ✓ Faster R-CNN에 mask head를 더해 Mask R-CNN 구조를 구축했듯이, **"DETR의 decoder 결과물에 mask head를 추가하여 segmentation task까지 확장**할 수 있다!"



* Panoptic Segmentation, A Kirillov et al., 2018