National Taiwan University
Computer Science & Information Engineering
Name: Cheng-Yu, Lin
ID: B10902024

113-1 (Fall 2024)
CVPDL

Assignment 1 Report

Q1. Draw the architecture of your object detector

In this assignment, I choose Relation-DETR as my model. Based on DETR, it introduces an explicit position relation prior, allowing the model to understand the spatial relationships between objects from the beginning, leading to faster and more efficient training.

Below is the architecture of DETR and the added positional relation in Relation-DETR.

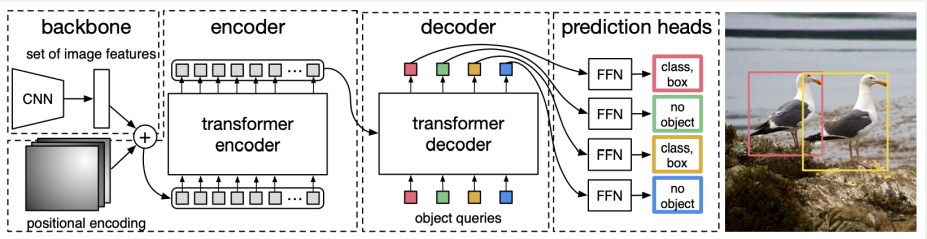


Figure 1: Architecture of DETR

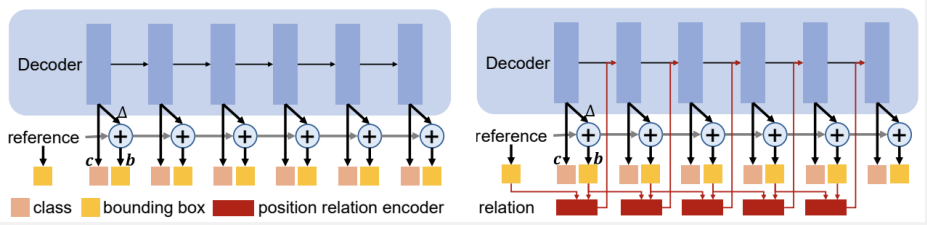


Figure 2: Comparison of transformer decoder in Deformable-DETR(left) and RelationDETR(right)

Q2. Implement details

1. Model and Backbone

   I choose transformer-based Relation-DETR as my model and using FocalNet-Large as backbone. The model weight is fine-tuned on COCO after pretrained on Object365. The implementation detail and pretrained weight can be found at `https://github.com/xiuqhou/Relation-DETR`.

2. Augmentation

```python
detr = T.Compose([
    T.RandomHorizontalFlip(),
    T.RandomChoice([
        T.RandomShortestSize(min_size=scales, max_size=1333, antialias=True),
        T.Compose([
            T.RandomShortestSize([400, 500, 600], antialias=True),
            RandomSizeCrop(384, 600),
            T.RandomShortestSize(min_size=scales, max_size=1333, antialias=True),
        ]),
    ]),
    T.PILToTensor(),
    T.ConvertImageDtype(torch.float),
    T.Normalize(mean=(0.485, 0.456, 0.406), std=(0.229, 0.224, 0.225)),
    T.SanitizeBoundingBox(labels_getter=labels_getter),
])
```

Figure 3: Transforms for augmentation

3. Loss function

   The loss function for Relation-DETR consists of two main parts, calculating the best match of predictions with respect to given ground truths using a graph technique with a cost function, and defining a loss to penalize the class and box predictions.

   For matching problem, the model utilizes the Hungarian algorithm to achieve best match. As for loss, Relation-DETR uses a combination of lossses, such as **Classification Loss**, **Bounding Box Loss**, and **GIoU Loss**. These losses are weighted and gathered to calculate the final loss.

4. Parameter settings

   - Learning rate: 1e-4
   - Batch size: 1
   - Epochs: 20
   - Mixed-precision: fp16
   - Maximum gradient norm: 0.1

   The major adjustment for parameter settings is **Batch size** and **Mixed-precision**. Since the original settings consume too much memory(over 40G), I lower the batch size from 2 to 1 and train with fp16 to reduce memory consumption.

Q3. Table of your performance for validation set

```
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.579
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.796
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.638
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.066
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.286
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.621
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.423
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.722
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.749
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.122
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.460
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.790
```

Figure 4: Performance metrics

class	imgs	gts	dets	recall	ap
Person	2045	3676	88465	0.983	0.958
Ear	1320	2080	111212	0.953	0.906
Earmuffs	24	2160	111212	0.938	0.732
Face	1511	2399	146533	0.988	0.959
Face-guard	8	2435	146533	0.944	0.703
Face-mask-medical	18	2568	146533	0.870	0.781
Foot	93	211	15513	0.919	0.622
Tools	610	1255	89379	0.905	0.651
Glasses	426	514	29735	0.944	0.838
Gloves	362	752	44008	0.967	0.860
Helmet	133	251	14281	0.960	0.887
Hands	1713	4201	82993	0.976	0.942
Head	1736	3223	72168	0.978	0.953
Medical-suit	26	31	10087	0.903	0.512
Shoes	440	1257	30084	0.955	0.863
Safefy-suit	43	70	13621	0.957	0.601
Safefy-vest	64	131	19866	1.000	0.763
mean results				0.949	0.796

Figure 5: Recall, AP under different classes

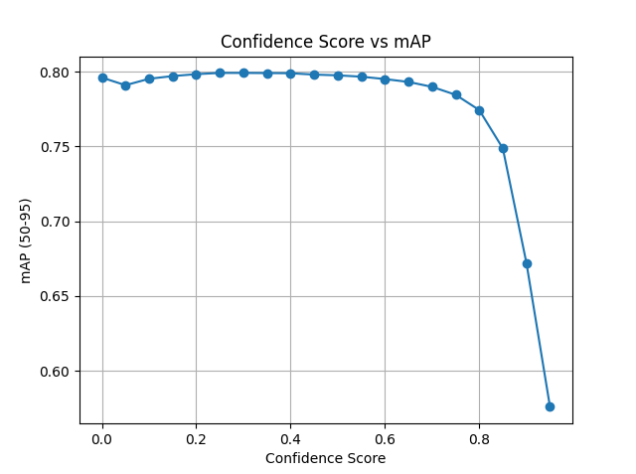


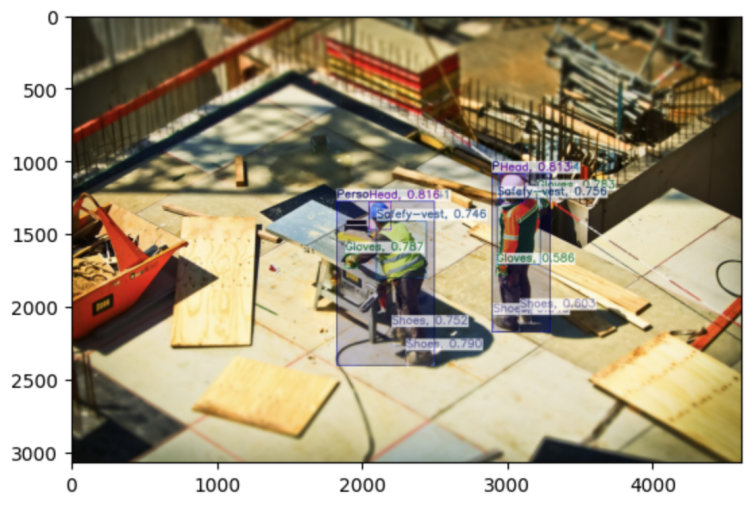
Figure 6: mAP with different confidence threshold

The results demonstrate that the model performs decently, especially on large area. Even though the dataset suffers from class imbalance, the model still retains AP over 0.5 for all different classes.

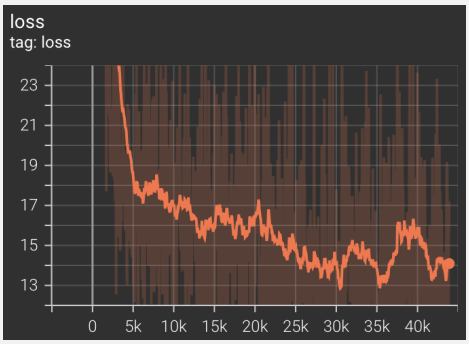
In order to set a proper threshold of the confidence score for prediction, I ran through all possibilities from 0 to 1 incremented by 0.05. The result illustrates that there's no major difference between different threshold under 0.6. The reason is that the model tends to predict the score either larger than 0.6 or smaller than 0.1, so setting different threshold won't affect the performance greatly. Eventually I decide to set the threshold to 0.3 to predict on test data.

Q4. Visualization and discussion

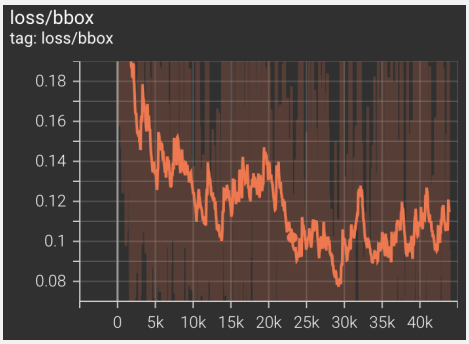
1. Visualization



2. Discussion

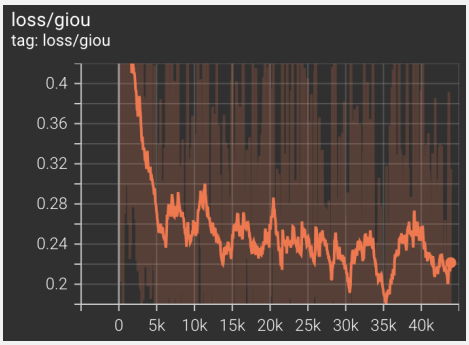


(a) Total loss



(b) Bounding box loss

(c) Classification loss



(d) GIoU loss

In the loss curve we can observe that Relation-DETR has extremely fast convergence speed. 2160 training steps is equivalent to an epoch, we can see that the model converges around the 3rd epoch, then suffer from the **Long Tail Effect**. One possible reason for the long tail effect is the class imbalance in the training data. As seen in 8, Person, Hands, and Head have thousands of instances in the training data while Medical-suit, Safety-suit, and Safety-vest only have hundreds. This causes the model to easily converge, but difficult to achieve high AP on certain classes.

class	imgs	gts	dets	recall	ap
Person	4057	7392	181367	0.987	0.970
Ear	2577	4038	226917	0.970	0.941
Earmuffs	64	4227	226917	0.995	0.939
Face	2980	4696	294498	0.994	0.976
Face-guard	14	4770	294498	1.000	0.945
Face-mask-medical	43	5046	294498	0.969	0.919
Foot	190	399	30799	0.955	0.844
Tools	1192	2469	170593	0.953	0.892
Glasses	839	1033	59999	0.969	0.914
Gloves	705	1509	85026	0.969	0.928
Helmet	240	522	29105	0.925	0.884
Hands	3458	8437	169625	0.984	0.971
Head	3445	6335	152165	0.982	0.969
Medical-suit	58	83	18521	1.000	0.982
Shoes	810	2347	56817	0.965	0.931
Safety-suit	70	125	24872	0.984	0.824
Safety-vest	104	302	37997	0.884	0.764
mean results				0.970	0.917

Figure 8: Recall, AP for training data