## 3.

The first part, I will give an introduction to object detection

## 4.

in the field of computer vision, object detection is a fundamental task

The goal of object detection is to predict a set of bounding boxes and category labels for each object of interest.

Modern detectors address this set prediction task in an indirect way, by defining surrogate regression and classification problems on a large set of proposals, anchors, or window centers.

~~For example, yolo(1) resizes the input image to 448 × 448, (2) runs a single convolutional network on the image, and (3) thresholds the resulting detections by the model’s confidence.~~

In the process, Anchors are used to generate the initial box in the regression task，

and NMS is used to remove the duplicate candidate boxes of the detection task, leaving only the candidate box with the highest prediction probability value as the final prediction result

The pictures show the results that YOLO running on sample artwork and natural images from the internet.

It is mostly accurate although it does think one person is an airplane.

DETR present a new method that views object detection as a direct set prediction problem.

The approach streamlines the detection pipeline, effectively removing the need for many hand-designed components like a non-maximum suppression procedure or anchor generation

~~that explicitly encode our prior knowledge about the task.~~

This is the baseline of the paper the presentation introduce today.

## 5.

Secondly, we will revisit detection transformer and deformable convolution.

## 6.

A priori knowledge, or baseline, is detr, which has the structure shown in the figure。

1.Introduction

DETR uses a conventional CNN backbone to learn a 2D representation of an input image.

The model flattens it and supplements it with a positional encoding before passing it into a transformer encoder.

A transformer decoder then takes as input a small fixed number of learned positional embeddings, which we call object queries, and additionally attends to the encoder output.

We pass each output embedding of the decoder to a shared feed forward network (FFN) that predicts either a detection (class and bounding box) or a “no object” class.

2.Defict

Despite its interesting design and good performance, DETR has its own issues:

(1) It requires much longer training epochs to converge than the existing object detectors.

~~For example, on the COCO benchmark, DETR needs 500 epochs to converge, which is around 10 to 20 times slower than Faster R-CNN .~~

(2) DETR delivers relatively low performance at detecting small objects.

Modern object detectors usually exploit multi-scale features, where small objects are detected from high-resolution feature maps.

Meanwhile, high-resolution feature maps lead to unacceptable complexities for DETR.

~~The above-mentioned issues can be mainly attributed to the deficit of Transformer components in processing image feature maps.~~

Therefore, In the subsequent lecture, I will present the improvement solution for this problem in the thesis

## 7.

1. Problem

In the image domain, deformable convolution is of a powerful and efficient mechanism to attend to sparse spatial locations.

It naturally avoids the issues mentioned in the previous slide.

While it lacks the element relation modeling mechanism, which is the key for the success of DETR.

2.Introduction

Here we introduce DC briefly .

The left picture is an illustration of 3 × 3 standard and deformable convolution。

It adds 2D offsets to the regular grid sampling locations in the standard convolution.

~~It enables free form deformation of the sampling grid.~~

The offsets are learned from the ~~preceding~~ feature maps, via additional convolutional layers.

~~Thus, the deformation is conditioned on the input features in a local, dense, and adaptive manner.~~

~~In the figure~~

~~regular sampling grid (green points) of standard convolution.~~

~~deformed sampling locations (dark blue points) with augmented offsets (light blue arrows) in deformable convolu-tion.~~

(c)(d) are special cases of (b), showing that the deformable convolution generalizes various transformations for scale, (anisotropic) aspect ratio and rotation.

The right picture illustrates of the fixed receptive field in standard convolution (a) and the adaptive receptive field in deformable convolution (b), using two layers.

~~Top: two activation units on the top feature map, on two objects of different scales and shapes. The activation is from a 3 × 3 filter.~~

~~Middle: the sampling locations of the 3 × 3 filter on the preceding feature map. Another two activation units are highlighted.~~

~~Bottom: the sampling locations of two levels of 3 × 3 filters on the preceding feature map. Two sets of locations are highlighted, corresponding to the highlighted units above.~~

Inspired by deformable convolution, the paper proposed deformable attention module.

## 8.

Thirdly, this is the introduction to method.

Here we mainly illustrate the main idea of the paper, but it also uses additional improvements, such as iterative bounding box refinement and two-stage paradigm.

## 9.

The proposed Deformable DETR, which mitigates the slow convergence and high complexity issues of DETR.

It combines the best of the sparse spatial sampling of deformable convolution, and the relation modeling capability of Transformers.

We propose the deformable attention module, which attends to a small set of sampling locations as a pre-filter for prominent key elements out of all the feature map pixels.

The module can be naturally extended to aggregating multi-scale features, without the help of FPN.

In Deformable DETR , we utilize (multi-scale) deformable attention modules to replace the Transformer attention modules processing feature maps.

## 10.

Let q ∈ Ωq indexes a query element with representation feature zq ~~∈ RC~~, and k ∈ Ωk indexes a key element with representation feature

xk ~~∈ RC, where C is the feature dimension, Ωq and Ωk specify the set of query and key elements, respectively.~~

m indexes the attention head

W is of learnable weights

The attention weights Amqk

~~an input feature map x ∈ RC×H×W , let q index a query element with content feature zq and a 2-d reference point pq,~~

~~the deformable attention feature is calculated by the second equation.~~

~~m indexes the attention head, k indexes the sampled keys, and K is the total sampled key number~~

∆pmqk and Amqk denote the sampling offset and attention weight of the kth sampling point in the mth attention head, respectively.

Both ∆pmqk and Amqk are obtained via linear projection over the query feature zq.

Let {xl}Ll=1 be the input multi-scale feature maps, ~~where xl ∈ RC×Hl×Wl.~~

Let ˆpq ~~∈ [0, 1]2~~ be the normalized coordinates of the reference point for each query element q, then the multi-scale deformable attention module is applied as the third equation.

Function φl( ˆpq) in Equation 3 re-scales the normalized coordinates ˆpq to the input feature map of the l-th level.

## 11.

Then this is the illustration of the proposed deformable attention module.

the deformable attention module only attends to a small set of key sampling points around a reference point, regardless of the spatial size of the feature maps, as shown in Fig.

In the figure K=3, M=3

By assigning only a small fixed number of keys for each query, the issues of convergence and feature spatial resolution can be mitigated.

Because the multi-scale deformable attention module extracts image features around the reference point,

~~we let the detection head predict the bounding box as relative offsets w.r.t. the reference point to further reduce the optimization difficulty.~~

The reference point is used as the initial guess of the box center.

The detection head predicts the relative offsets w.r.t. the reference point.

In this way, the learned decoder attention will have strong correlation with the predicted bounding boxes, which also accelerates the training convergence.

## 12.

1.The left figure is from Detr.

It visualizes decoder attention for every predicted object ..

~~Attention scores are coded with different colors for different objects.~~

From the pictures, we can get that decoder typically attends to object extremities, such as legs and heads.

2.The right figure shows what deformable detr looks at?

we draw the gradient norm of each item in final prediction

~~(i.e., x/y coordinate of object center, width/height of object bounding box, category score of this object) with respect to each pixel in the image.~~

~~According to Taylor’s theorem, the gradient norm can reflect how much the output would be changed relative to the perturbation of the pixel,~~

~~thus it could show us which pixels the model mainly relys on for predicting each item.~~

Deformable DETR attends to left/right boundary of the object for x coordinate and width, and top/bottom boundary for y coordinate and height.

Meanwhile, different to DETR, our Deformable DETR also looks at pixels inside the object for predicting its category.

## 13.

The paper visualize sampling points and attention weights of the last layer in encoder and decoder as show in this slide.

~~It combine the sampling points and attention weights from feature maps of different resolutions into one picture.~~

Similar to DETR, the instances are already separated in the encoder of Deformable DETR.

While in the decoder, our model is focused on the whole foreground instance instead of only extreme points as observed in DETR .

Combined with the visualization of the gradient norm of category score of the object in the last figure,

we can guess the reason is that our Deformable DETR needs not only extreme points but also interior points to determine object category.

The visualization demonstrates that the proposed multi-scale deformable attention module can adapt its sampling points and attention weights according to different scales and shapes of the foreground object.

## 14.

## 15.

1.

The upper figure is convergence curves.

~~For Deformable DETR, we explore different training schedules by varying the epochs at which the learning rate is reduced (where the AP score leaps).~~

Compared with DETR, Deformable DETR achieves better performance (especially on small objects) with 10× less training epochs

### ~~DETR-DC5: increase the feature resolution by adding a dilation to the last stage of the backbone and removing a stride from the first convolution of this stage.~~

~~Our proposed Deformable DETR has on par FLOPs with other networks.~~

~~But the runtime speed is much faster.~~

~~The speed issue of DETR-DC5 is mainly due to the large amount of memory access in Transformer attention.~~

~~Our proposed deformable attention can mitigate this issue, at the cost of unordered memory access.~~

Thus, it is still slightly slower than traditional convolution.

With the aid of iterative bounding box refinement and two-stage paradigm, our method can further improve the detection accuracy.

## 16.

1.

The upper table shows the results of ablation study.

Using multi-scale inputs instead of single-scale inputs can effectively improve detection accuracy with 1.7% AP , especially on small objects with 2.9% APS.

Increasing the number of sampling points K can further improve 0.9% AP .

Using multi-scale deformable attention, which allows information exchange among different scale levels, can bring additional 1.5% improvement in AP .

Because the cross-level feature exchange is already adopted, adding FPNs will not improve the performance.

~~When multi-scale attention is not applied, and K = 1, our (multi-scale) deformable attention module degenerates to deformable convolution, delivering noticeable lower accuracy.~~

2

The lower table compares the proposed method with other state-of-the-art methods.

~~Iterative bounding box refinement and two-stage mechanism are both utilized by our models in the table.~~

~~With ResNet-101 and ResNeXt-101, our method achieves 48.7 AP and 49.0 AP without bells and whistles, respectively.~~

By using ResNeXt-101 with DCN, the accuracy rises to 50.1 AP.

With additional test-time augmentations, the proposed method achieves 52.3 AP.

## 17.

## 18.

Deformable DETR is an end-to-end object detector, which is efficient and fast-converging.

It enables us to explore more interesting and practical variants of end-to-end object detectors.

At the core of Deformable DETR are the (multi-scale) deformable attention modules,

which is an efficient attention mechanism in processing image feature maps.

The work opens up new possibilities in exploring end-to-end object detection.