

# Transformers for Image Recognition

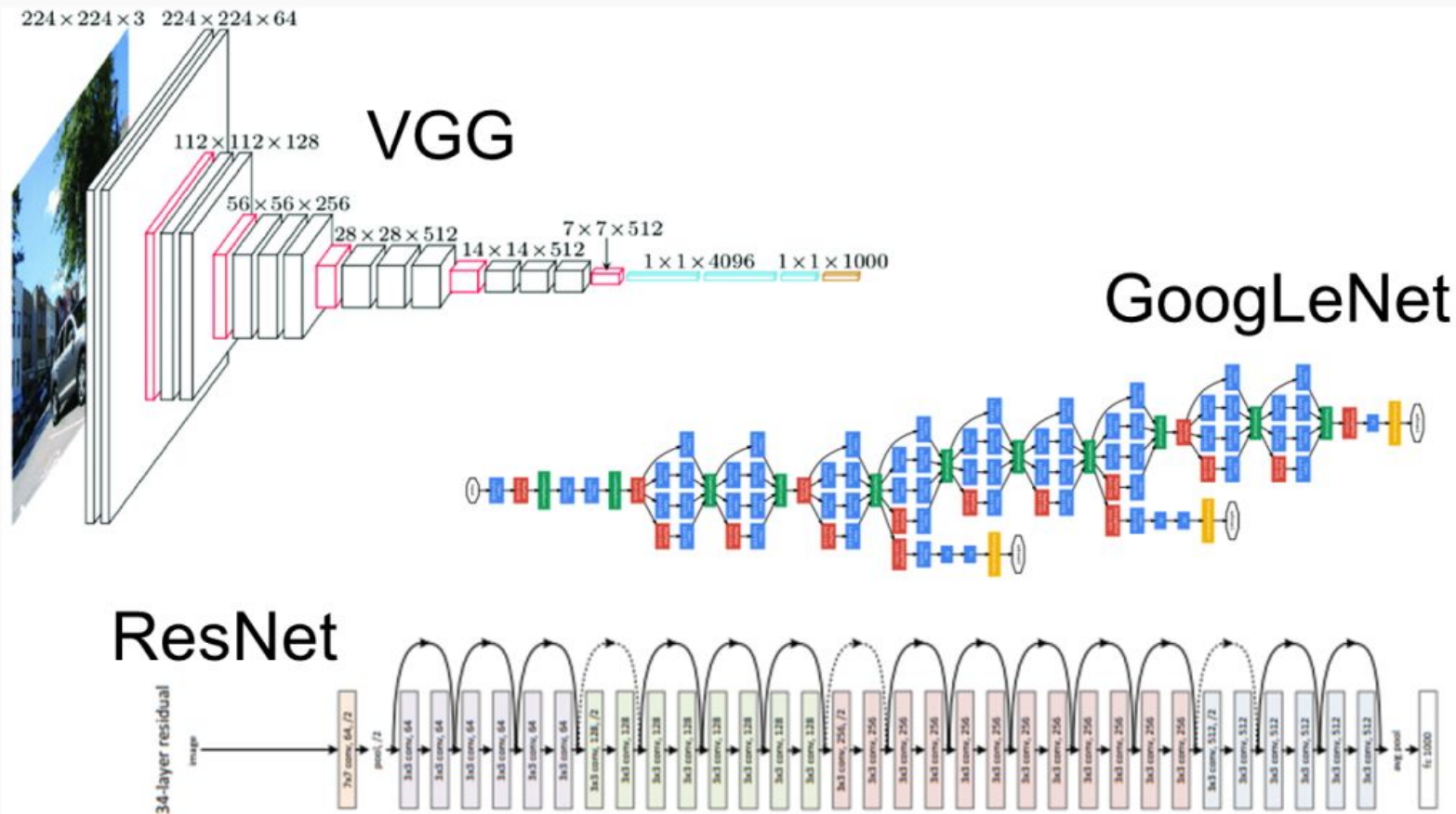
Никита Андреев



# План

- Идея использования трансформеров
- Vision Transformer
- DEtection TRansformer

# Зачем трансформеры?



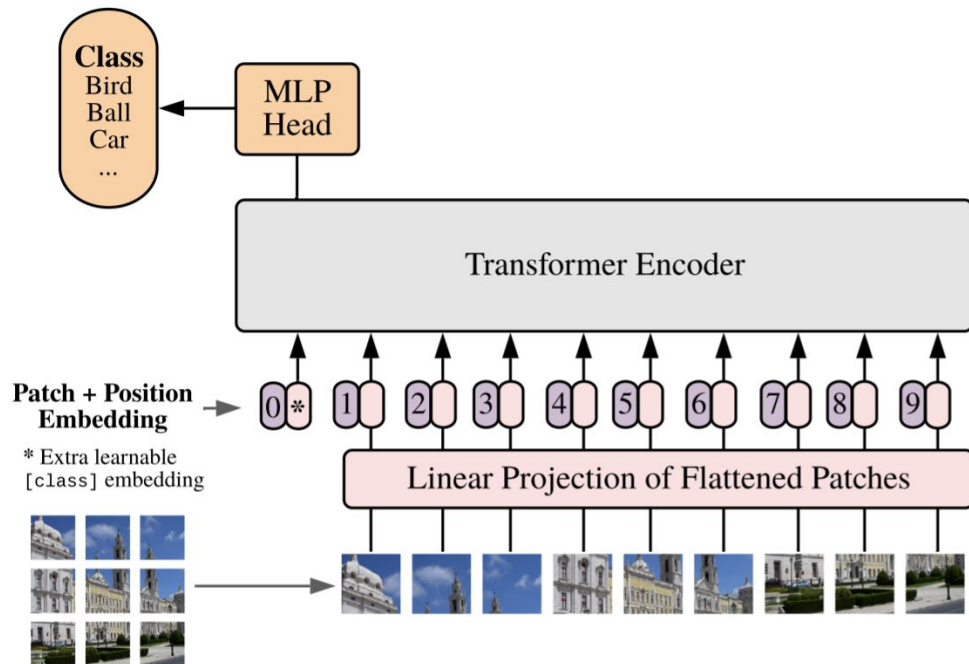
# Как применить

- Попиксельный attention
- Локальный attention только для соседних пикселей
- Другие различные аппроксимации

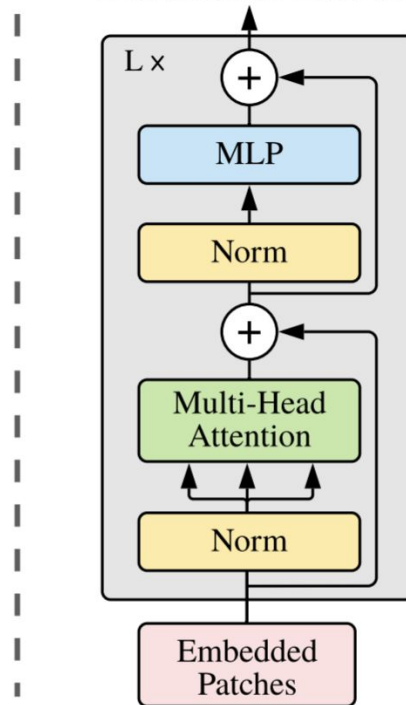
Результаты - обнадеживающие, но довольно трудоемкие алгоритмы для эффективной реализации.

# Vision Transformer

## Vision Transformer (ViT)



## Transformer Encoder



Model	Layers	Hidden size $D$	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

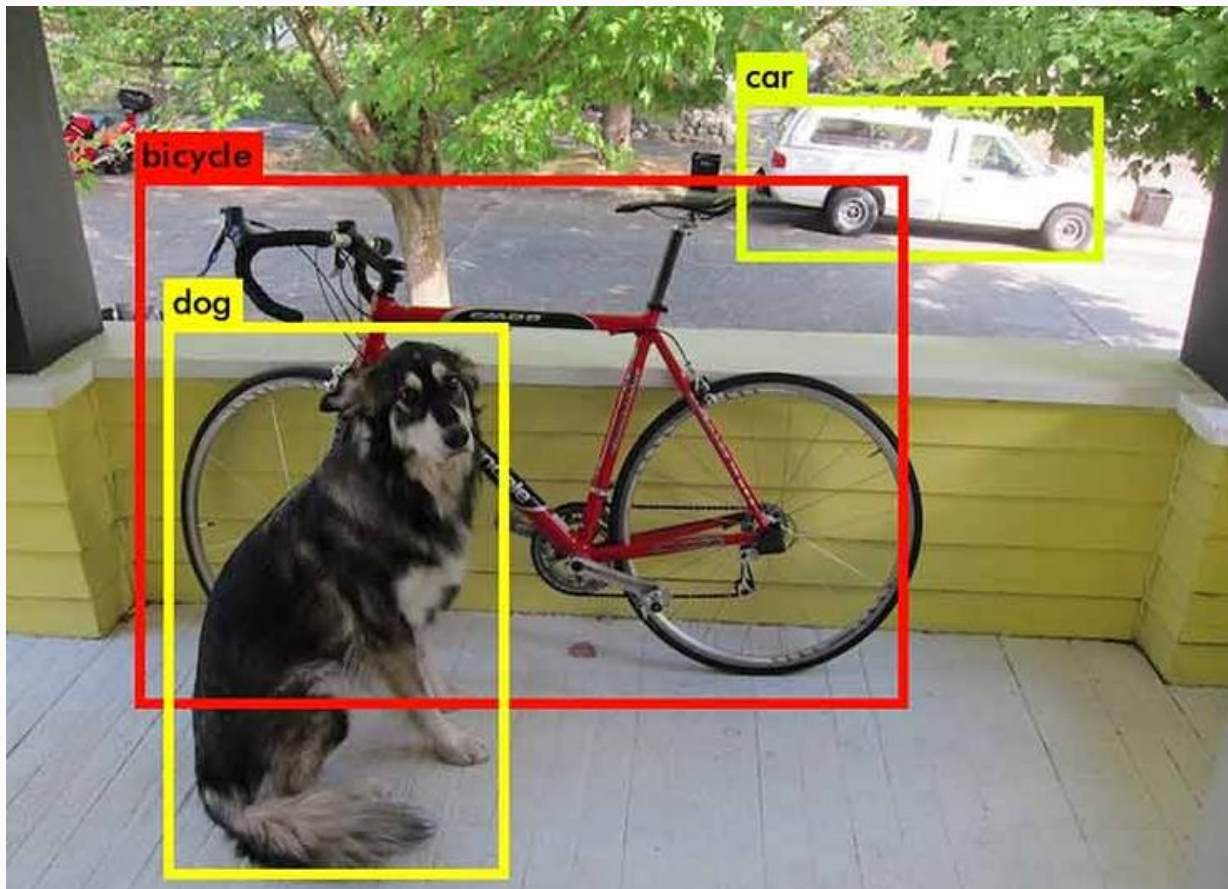
Table 1: Details of Vision Transformer model variants.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21K (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	<b>88.55</b> $\pm 0.04$	87.76 $\pm 0.03$	85.30 $\pm 0.02$	87.54 $\pm 0.02$	88.4/88.5*
ImageNet ReaL	<b>90.72</b> $\pm 0.05$	90.54 $\pm 0.03$	88.62 $\pm 0.05$	90.54	90.55
CIFAR-10	<b>99.50</b> $\pm 0.06$	99.42 $\pm 0.03$	99.15 $\pm 0.03$	99.37 $\pm 0.06$	—
CIFAR-100	<b>94.55</b> $\pm 0.04$	93.90 $\pm 0.05$	93.25 $\pm 0.05$	93.51 $\pm 0.08$	—
Oxford-IIIT Pets	<b>97.56</b> $\pm 0.03$	97.32 $\pm 0.11$	94.67 $\pm 0.15$	96.62 $\pm 0.23$	—
Oxford Flowers-102	99.68 $\pm 0.02$	<b>99.74</b> $\pm 0.00$	99.61 $\pm 0.02$	99.63 $\pm 0.03$	—
VTAB (19 tasks)	<b>77.63</b> $\pm 0.23$	76.28 $\pm 0.46$	72.72 $\pm 0.21$	76.29 $\pm 1.70$	—
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

# Object Detection

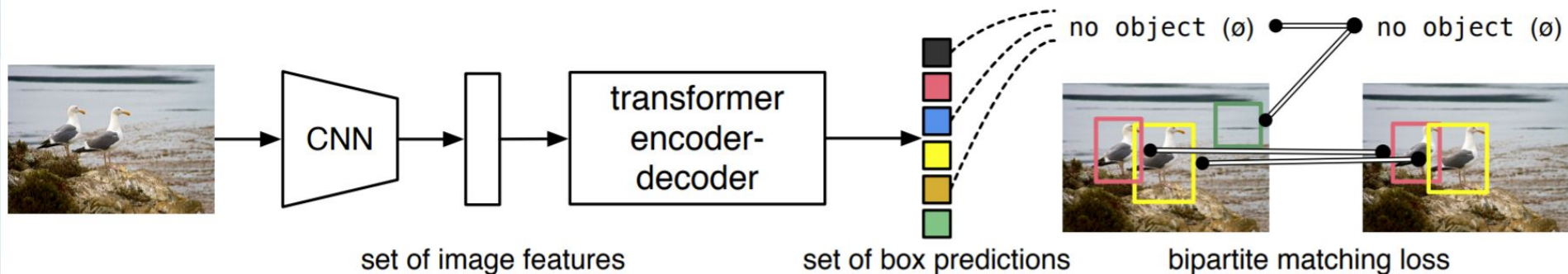


# Задача Object Detection



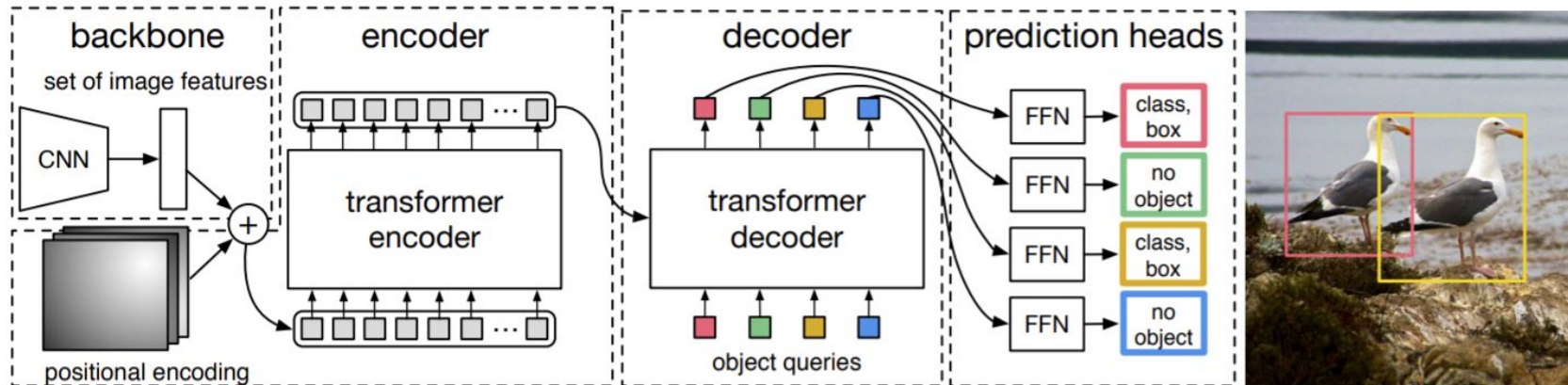
Хотим выделить на изображении объекты интересующих нас классов

# DEtection TRansformer



- Предобученная CNN
- Карта признаков изображения
- Трансформер(encoder + decoder)
- Множество предсказанных объектов
- Посчитать лосс

# Архитектура DETR



- Добавляем positional encoding
- На вход для декодера подаем фиксированное число запросов
- Параллельное декодирование для запросов
- Для каждого выхода получаем предсказание через FFN

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

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

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

$$\lambda_{\text{iou}} \mathcal{L}_{\text{iou}}(b_i, \hat{b}_{\sigma(i)}) + \lambda_{\text{L1}} \|b_i - \hat{b}_{\sigma(i)}\|_1$$

# Результаты

Model	GFLOPS/FPS	#params	AP	AP <sub>50</sub>	AP <sub>75</sub>	AP <sub>S</sub>	AP <sub>M</sub>	AP <sub>L</sub>
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	<b>47.8</b>	<b>27.2</b>	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	<b>44.9</b>	<b>64.7</b>	47.7	23.7	<b>49.5</b>	<b>62.3</b>

## Использованные материалы

1. <https://arxiv.org/pdf/2005.12872.pdf>
2. <https://arxiv.org/pdf/2101.01169.pdf>
3. <https://arxiv.org/pdf/2010.11929.pdf>