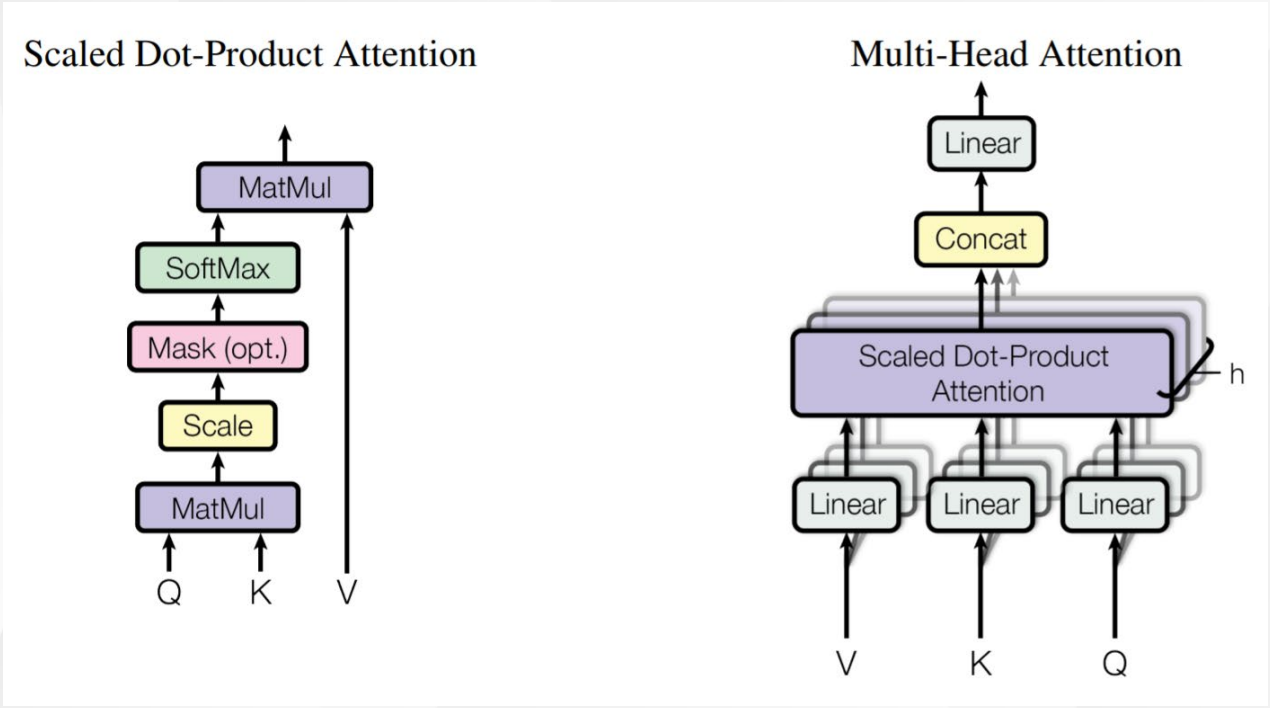
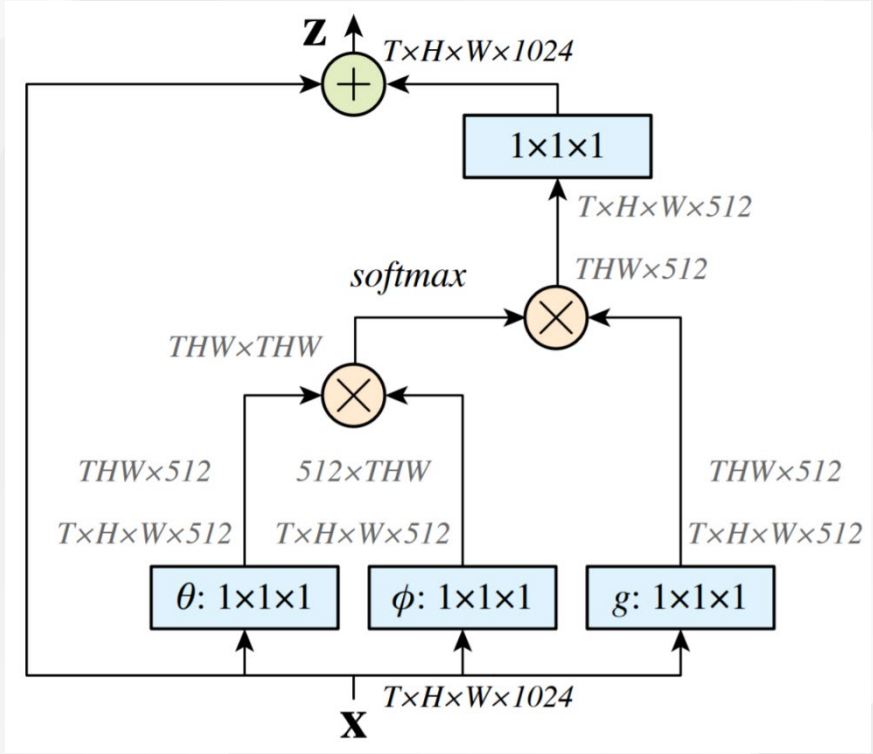
Content

- Introduction to Computer Vision
- Traditional Neural Networks and Their Limitations
- CNN and Recent Progress
  - CNN
  - Rectified Linear Units (ReLU)
  - Dropout / Batch Normalization
  - Representative Network Architectures
- Vision Transformers and Recent Progress

Spatial Self-Attention

- Self-Attention Multi-Head Attention Non-Local Block



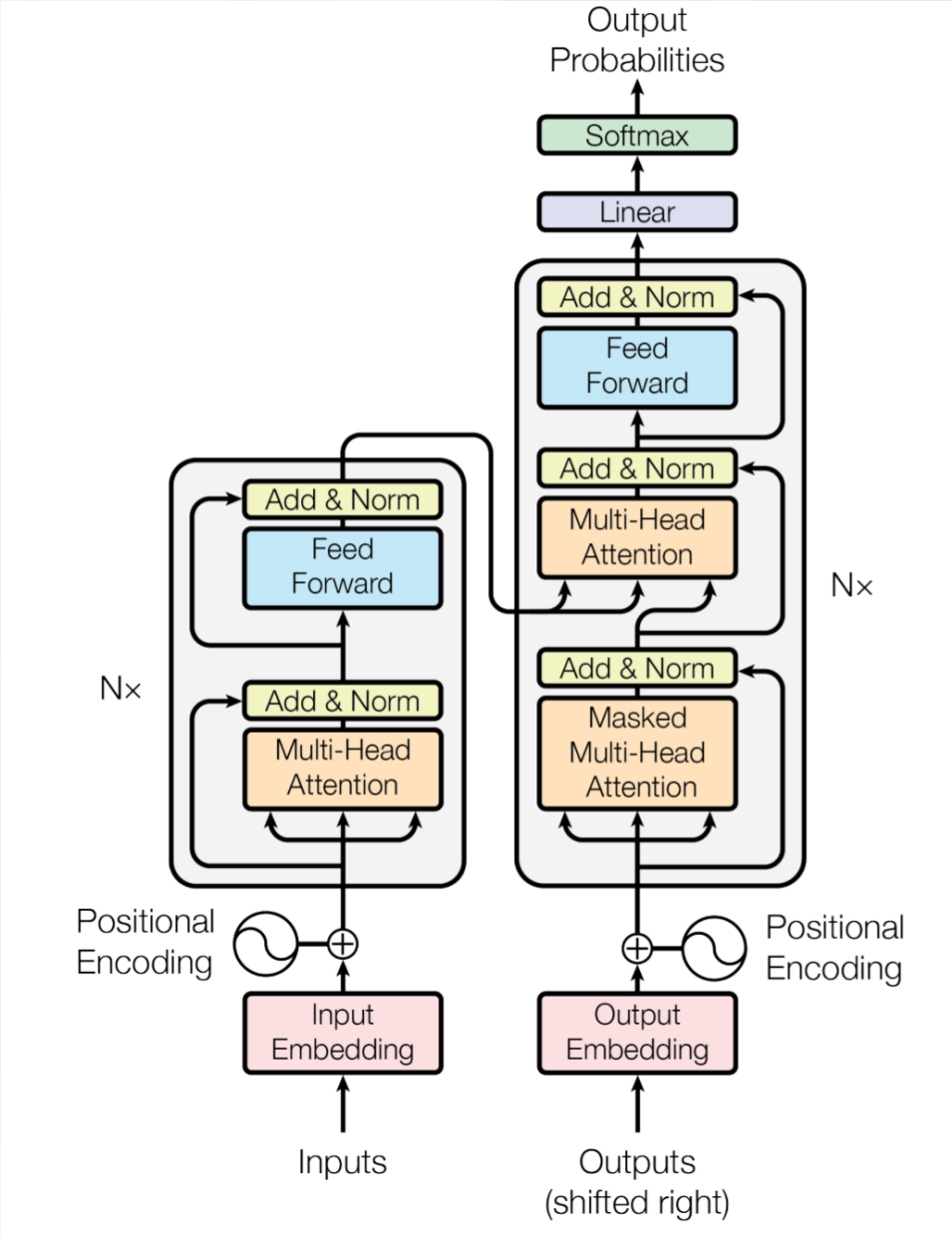


Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. NeurIPS, 2017.

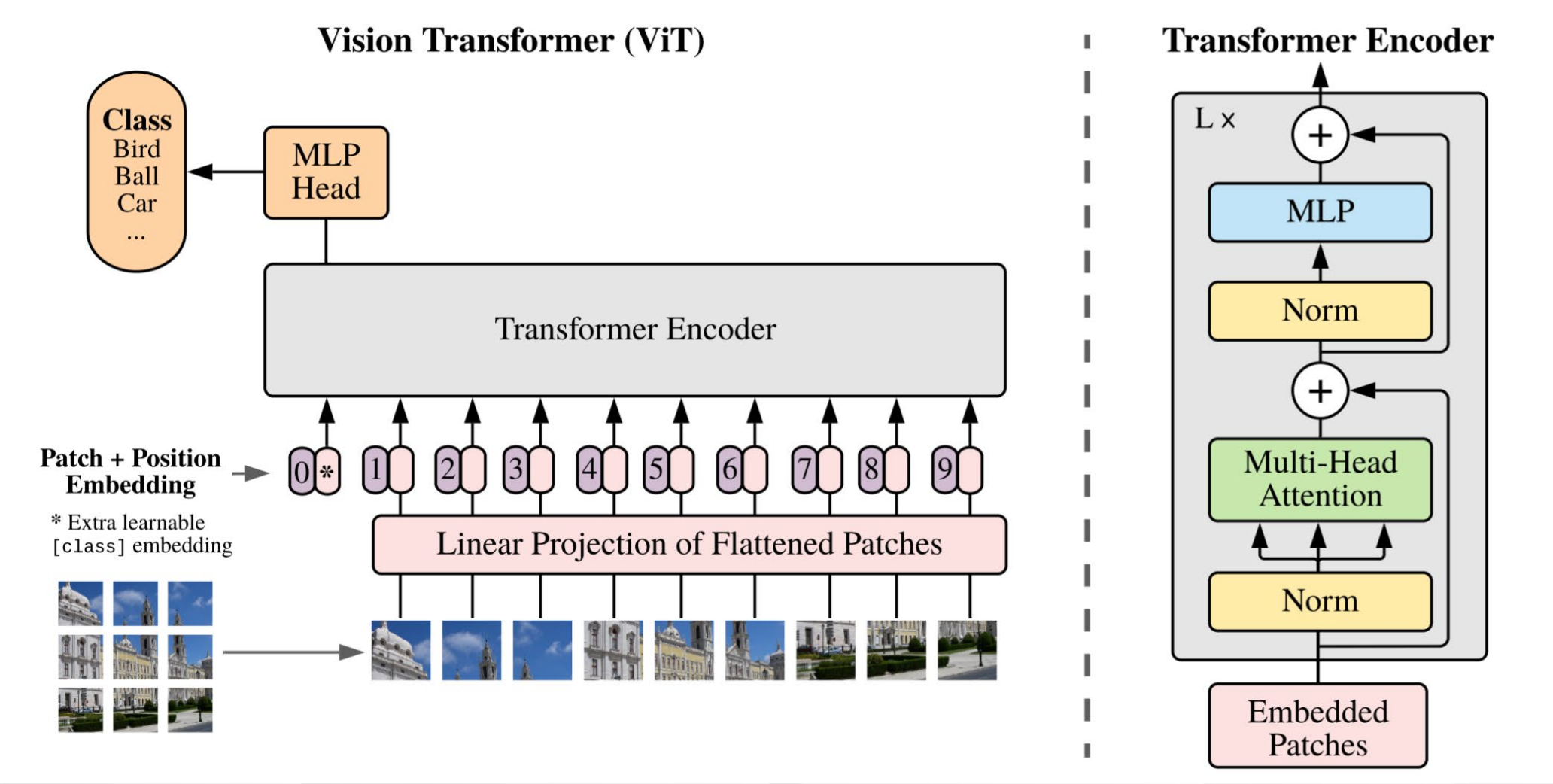
Transformer

- NLP
- Encoder-decoder
- Feed Forward Network

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$



Vision Transformer



*Tokens-to-Token ViT: Training Vision Transformers from Scratch on ImageNet, Arxiv 2020

DeiT

- Grid Search on Optimization Algorithms, Hyper-parameters, and Data Augmentation

Ablation on ↓	Pre-training	Fine-tuning	Rand-Augment	AutoAug	Mixup	CutMix	Erasing	Stoch. Depth	Repeated Aug.	Dropout	Exp. Moving Avg.	top-1 accuracy: pre-trained 224^2	top-1 accuracy: fine-tuned 384^2
none: DeiT-B	adamw	adamw	✓	✗	✓	✓	✓	✓	✓	✗	✗	81.8 ±0.2	83.1 ±0.1
optimizer	SGD	adamw	✓	✗	✓	✓	✓	✓	✓	✗	✗	74.5	77.3
	adamw	SGD	✓	✗	✓	✓	✓	✓	✓	✗	✗	81.8	83.1
data augmentation	adamw	adamw	✗	✗	✓	✓	✓	✓	✓	✗	✗	79.6	80.4
	adamw	adamw	✗	✓	✓	✓	✓	✓	✓	✗	✗	81.2	81.9
	adamw	adamw	✓	✗	✗	✓	✓	✓	✓	✗	✗	78.7	79.8
	adamw	adamw	✓	✗	✓	✗	✓	✓	✓	✗	✗	80.0	80.6
	adamw	adamw	✓	✗	✗	✗	✓	✓	✓	✗	✗	75.8	76.7
regularization	adamw	adamw	✓	✗	✓	✓	✗	✓	✓	✗	✗	4.3*	0.1
	adamw	adamw	✓	✗	✓	✓	✓	✗	✓	✗	✗	3.4*	0.1
	adamw	adamw	✓	✗	✓	✓	✓	✓	✗	✗	✗	76.5	77.4
	adamw	adamw	✓	✗	✓	✓	✓	✓	✓	✓	✗	81.3	83.1
	adamw	adamw	✓	✗	✓	✓	✓	✓	✓	✗	✓	81.9	83.1

Training data-efficient image transformers & distillation through attention, Arxiv 2020.

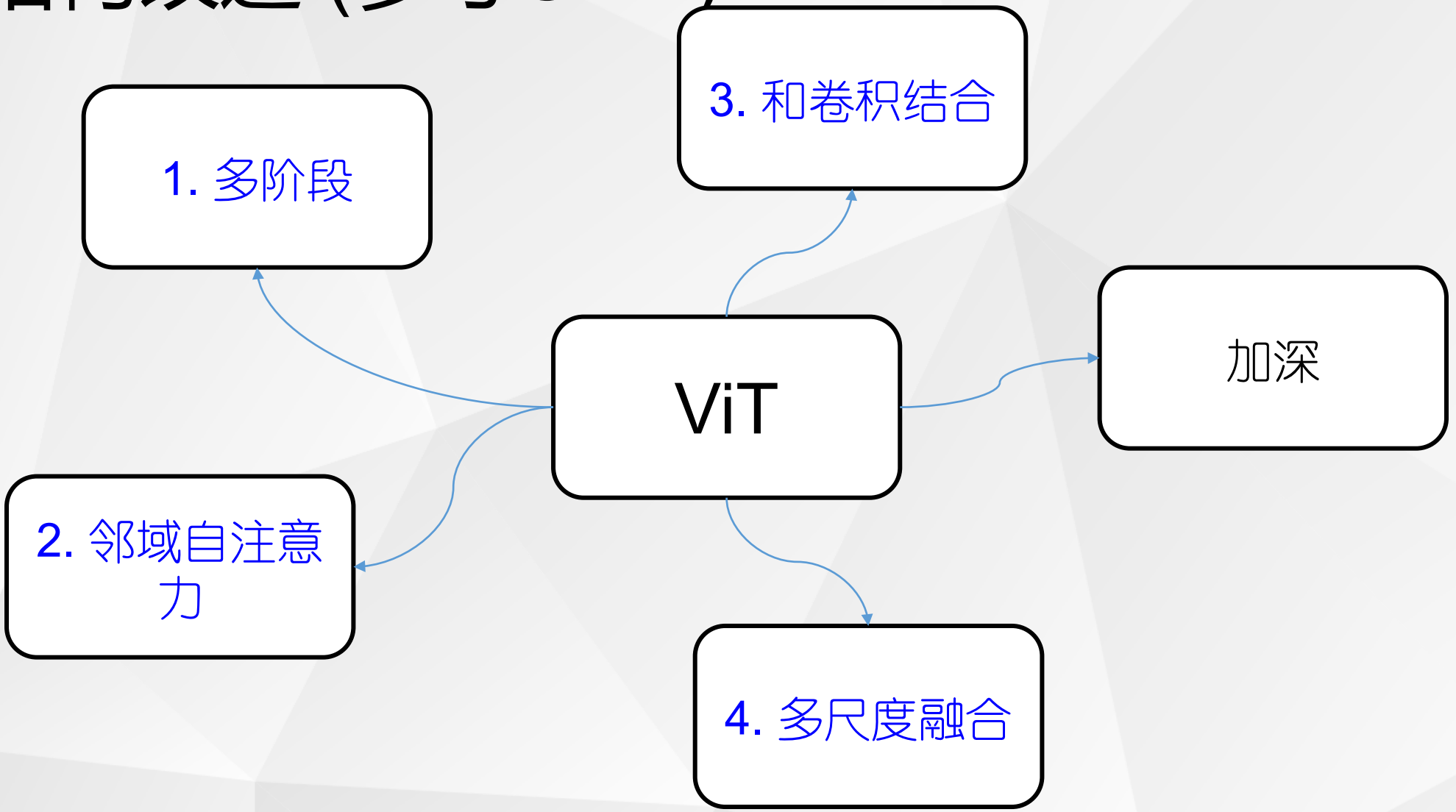
DeiT

- AdamW+strong regularization+ strong augmentation+more training epochs

Methods	ViT-B [15]	DeiT-B
Epochs	300	300
Batch size	4096	1024
Optimizer	AdamW	AdamW
learning rate	0.003	$0.0005 \times \frac{\text{batchsize}}{512}$
Learning rate decay	cosine	cosine
Weight decay	0.3	0.05
Warmup epochs	3.4	5
Label smoothing ε	✗	0.1
Dropout	0.1	✗
Stoch. Depth	✗	0.1
Repeated Aug	✗	✓
Gradient Clip.	✓	✗
Rand Augment	✗	9/0.5
Mixup prob.	✗	0.8
Cutmix prob.	✗	1.0
Erasing prob.	✗	0.25

Table 9: Ingredients and hyper-parameters for our method and Vit-B.

结构改进 (参考CNN)



1. Pyramid Vision Transformer

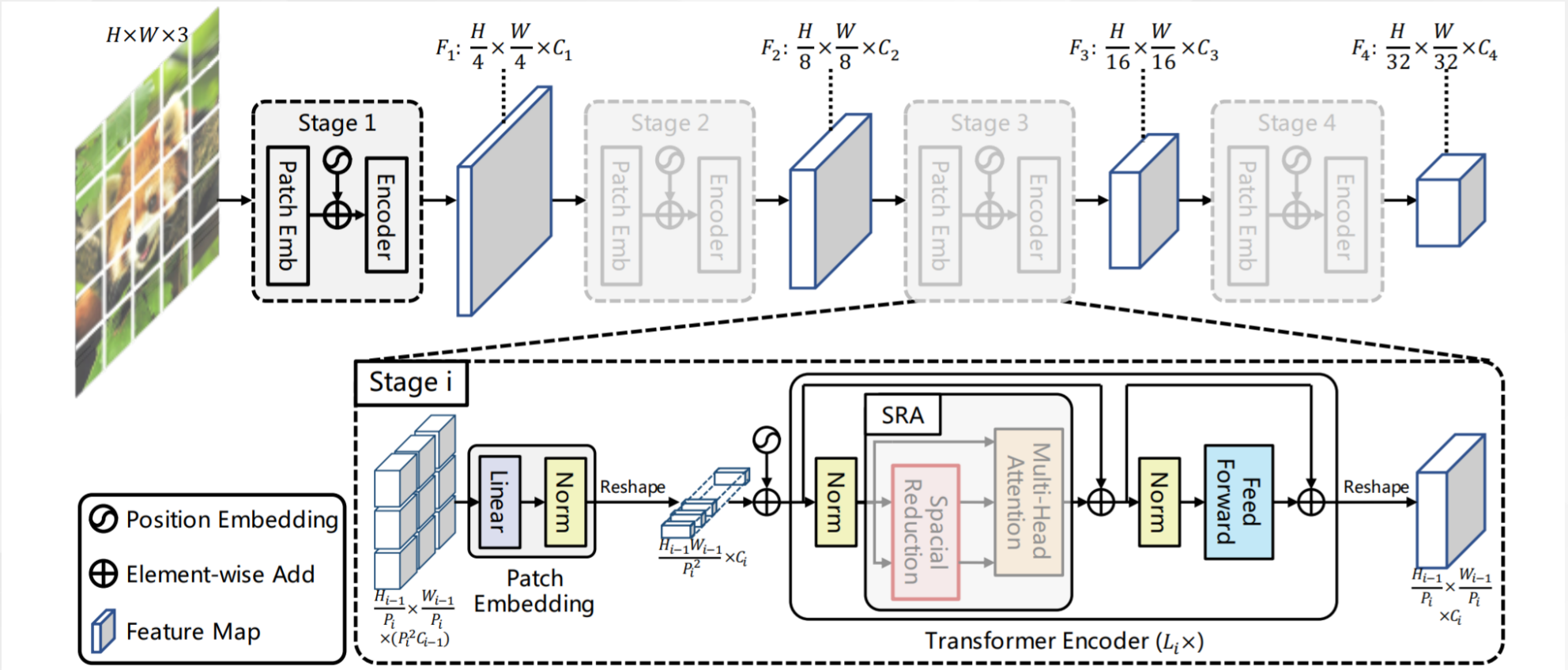


Figure 3: **Overall architecture of Pyramid Vision Transformer (PVT).** The entire model is divided into four stages, each of which is comprised of a patch embedding layer and a L_i-layer Transformer encoder. Following a pyramid structure, the output resolution of the four stages progressively shrinks from high (4-stride) to low (32-stride).

Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction Without Convolutions, ICCV 2021.

2. XCiT

- 不构造spatial self-attention
- 改成channel self-attention（空间信息交互用depthwise conv完成）

XCiT: Cross-Covariance Image Transformers, Arxiv 2021.

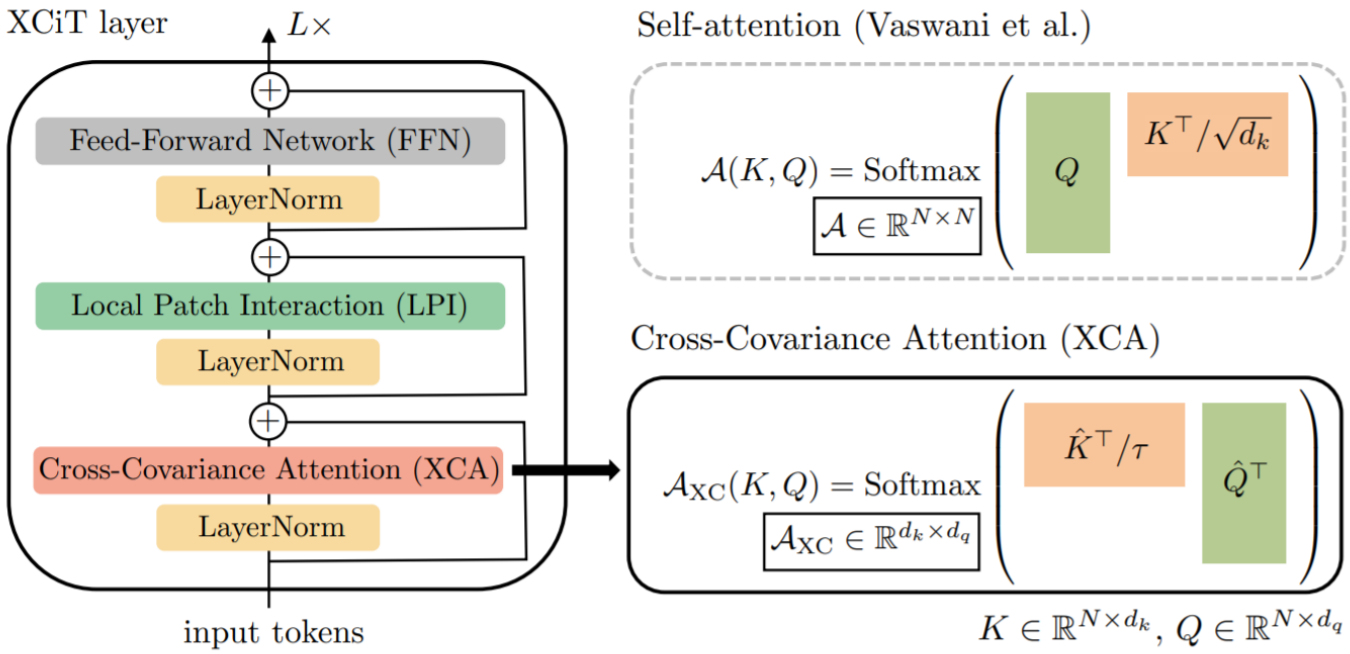


Figure 1: Our XCiT layer consists of three main blocks, each preceded by LayerNorm and followed by a residual connection: (i) the core cross-covariance attention (XCA) operation, (ii) the local patch interaction (LPI) module, and (iii) a feed-forward network (FFN). By transposing the query-key interaction, the computational complexity of XCA is linear in the number of data elements N, rather than quadratic as in conventional self-attention.

3. Early Convolutions

model	flops (B)	params (M)	acts (M)	time (min)	batch size	epochs 100	epochs 200	epochs 400	IN 21k
ResNet-50	4.1	25.6	11.3	3.4	2048	22.5	21.2	20.7	21.6
ResNet-101	7.8	44.5	16.4	5.5	2048	20.3	19.1	18.5	19.2
ResNet-152	11.5	60.2	22.8	7.7	2048	19.5	18.4	17.7	18.2
ResNet-200	15.0	64.7	32.3	10.7	1024	19.5	18.3	17.6	17.7
RegNetY-1GF	1.0	9.6	6.2	3.1	2048	23.2	22.2	21.5	-
RegNetY-4GF	4.1	22.4	14.5	7.6	2048	19.4	18.3	17.9	18.4
RegNetY-16GF	15.5	72.3	30.7	17.9	1024	17.1	16.4	16.3	15.6
RegNetY-32GF	31.1	128.6	46.2	35.1	512	16.2	15.9	15.9	15.0
RegNetZ-1GF	1.0	11.0	8.8	4.2	2048	20.8	20.2	19.6	-
RegNetZ-4GF	4.0	28.1	24.3	12.9	1024	17.4	16.9	16.6	-
RegNetZ-16GF	16.0	95.3	51.3	32.0	512	16.0	15.9	15.9	-
RegNetZ-32GF	32.0	175.1	79.6	55.3	256	16.3	16.2	16.1	-

model	flops (B)	params (M)	acts (M)	time (min)	batch size	epochs 100	epochs 200	epochs 400	IN 21k
EffNet-B2	1.0	9.1	13.8	5.9	2048	21.4	20.5	19.9	-
EffNet-B4	4.4	19.3	49.5	19.4	512	18.5	17.8	17.5	-
EffNet-B5	10.3	30.4	98.9	41.7	256	17.3	17.0	17.0	-
ViT$_P$-1GF	1.1	4.8	5.5	2.6	2048	33.2	29.7	27.7	-
ViT$_P$-4GF	3.9	18.5	11.1	3.8	2048	23.3	20.8	19.6	20.6
ViT$_P$-18GF	17.5	86.6	24.0	11.5	1024	19.9	18.4	17.9	16.4
ViT$_P$-36GF	35.9	178.4	37.3	18.8	512	19.9	18.8	18.2	15.1
ViT$_C$-1GF	1.1	4.6	5.7	2.7	2048	28.6	26.1	24.7	-
ViT$_C$-4GF	4.0	17.8	11.3	3.9	2048	20.9	19.2	18.6	18.8
ViT$_C$-18GF	17.7	81.6	24.1	11.4	1024	18.4	17.5	17.0	15.1
ViT$_C$-36GF	35.0	167.8	36.7	18.6	512	18.3	17.6	16.8	14.2

Table 2: **Peak performance (grouped by model family)**: Model complexity and validation top-1 error at 100, 200, and 400 epoch schedules on ImageNet-1k, and the top-1 error after pretraining on ImageNet-21k (IN 21k) and fine-tuning on ImageNet-1k. This table serves as reference for the results shown in Figure 6. Blue numbers: best model trainable under 20 minutes per ImageNet-1k epoch. Batch sizes and training times are reported normalized to 8 32GB Volta GPUs (see Appendix).

Early Convolutions Help Transformers See Better, Arxiv 2021

BoTNet

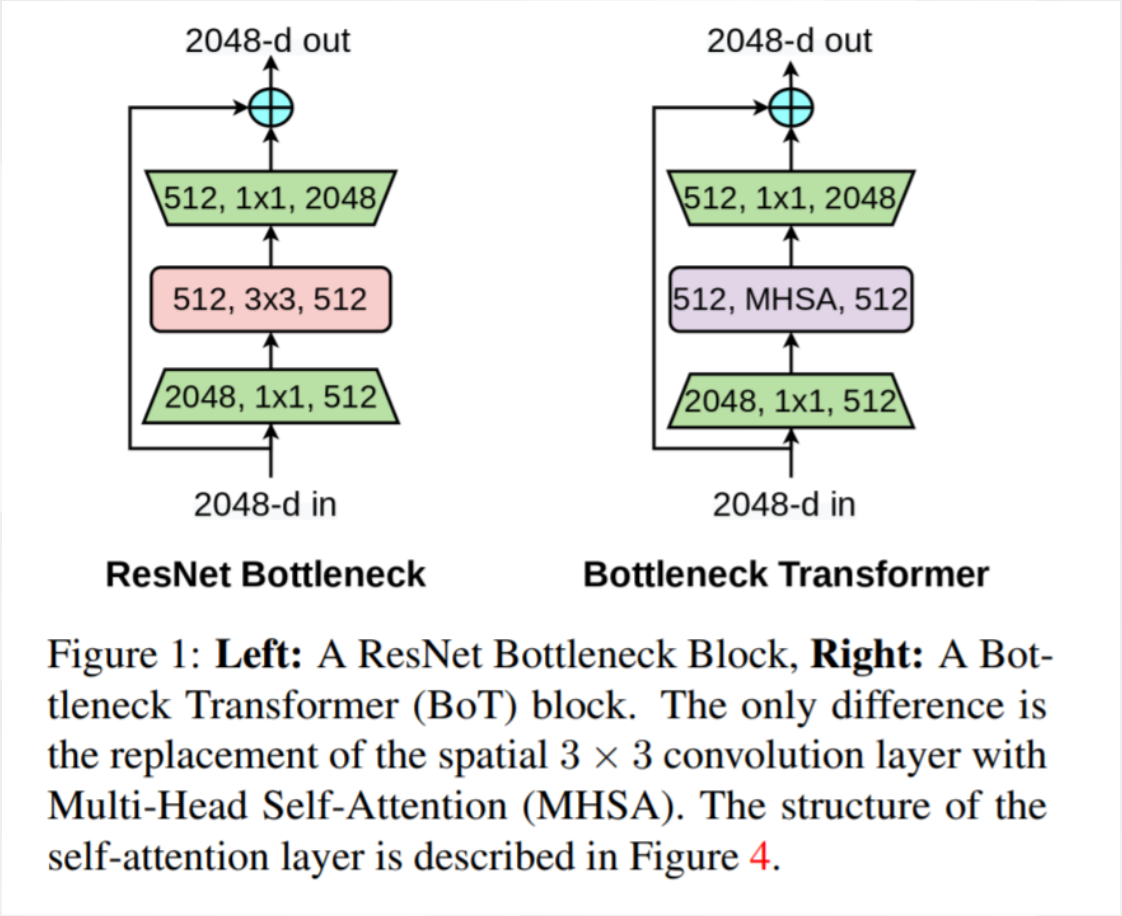


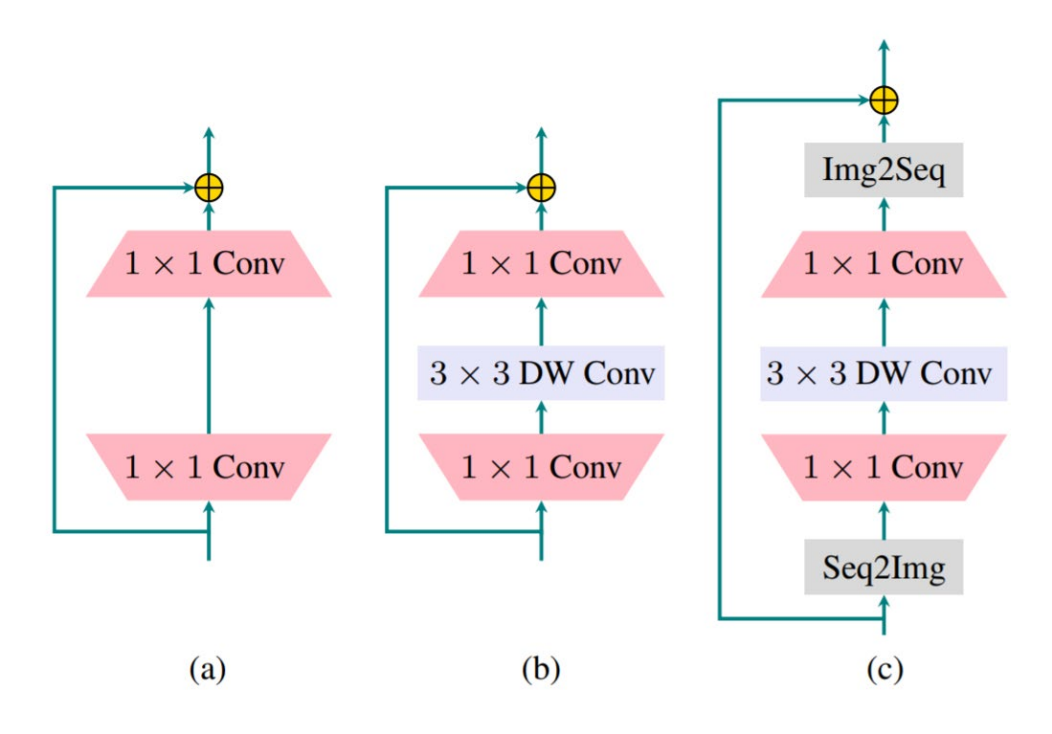
Figure 1: **Left:** A ResNet Bottleneck Block, **Right:** A Bottleneck Transformer (BoT) block. The only difference is the replacement of the spatial 3×3 convolution layer with Multi-Head Self-Attention (MHSA). The structure of the self-attention layer is described in Figure 4.

Bottleneck Transformers for Visual Recognition, Arxiv 2021.

LocalViT

- 将FFN替换成depthwise conv

Network	γ	DW	Params (M)	FLOPs (G)	Top-1 Acc. (%)
DeiT-T [41]	4	No	5.7	1.3	72.2
LocalViT-T	4	No	5.7	1.3	72.5 (0.3↑)
LocalViT-T*	4	Yes	5.8	1.3	73.7 (1.5↑)
DeiT-T [41]	6	No	7.5	1.6	73.1†
LocalViT-T	6	No	7.5	1.6	74.3 (1.2↑)
LocalViT-T*	6	Yes	7.7	1.6	76.1 (3.0↑)



LocalViT: Bringing Locality to Vision Transformers, Arxiv 2021.

4. 多尺度融合

- CoaT

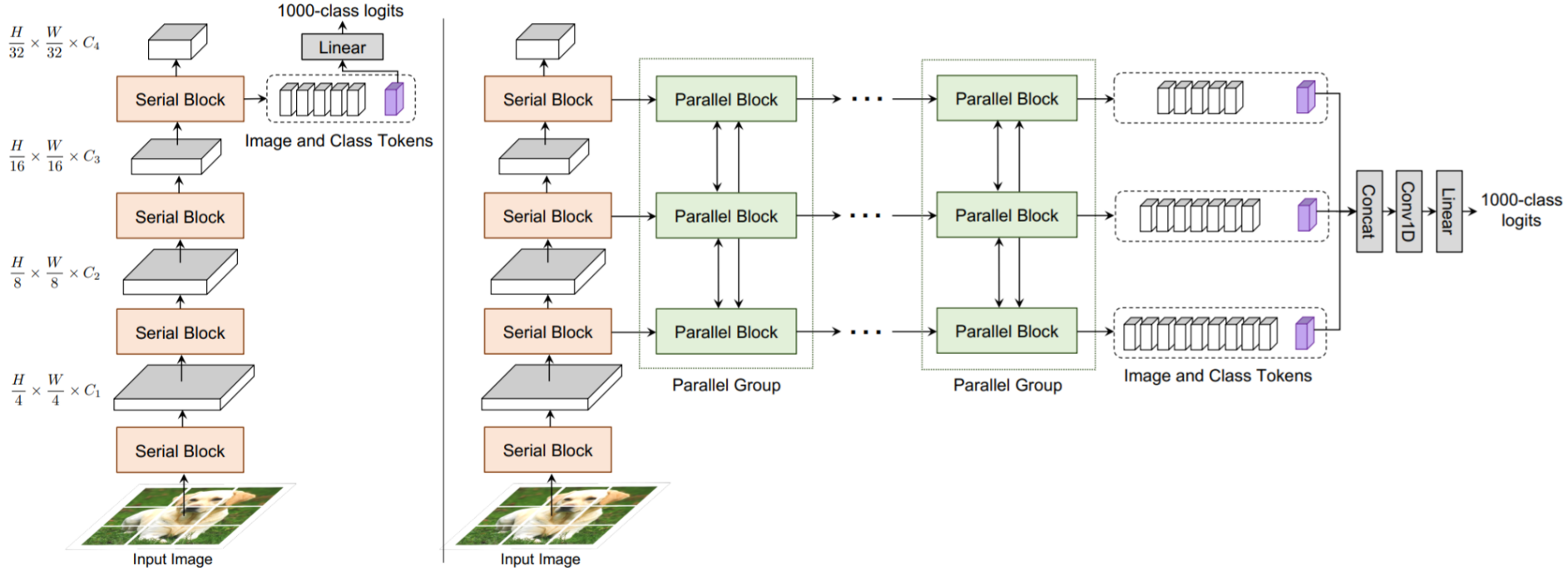


Figure 3. **CoaT model architecture.** (Left) The overall network architecture of **CoaT-Lite**. CoaT-Lite consists of serial blocks only, where image features are down-sampled and processed in a sequential order. (Right) The overall network architecture of **CoaT**. CoaT consists of serial blocks and parallel blocks. Both blocks enable the co-scale mechanism.

Co-Scale Conv-Attentional Image Transformers, Arxiv 2021.

Ensembling: Swin Transformer

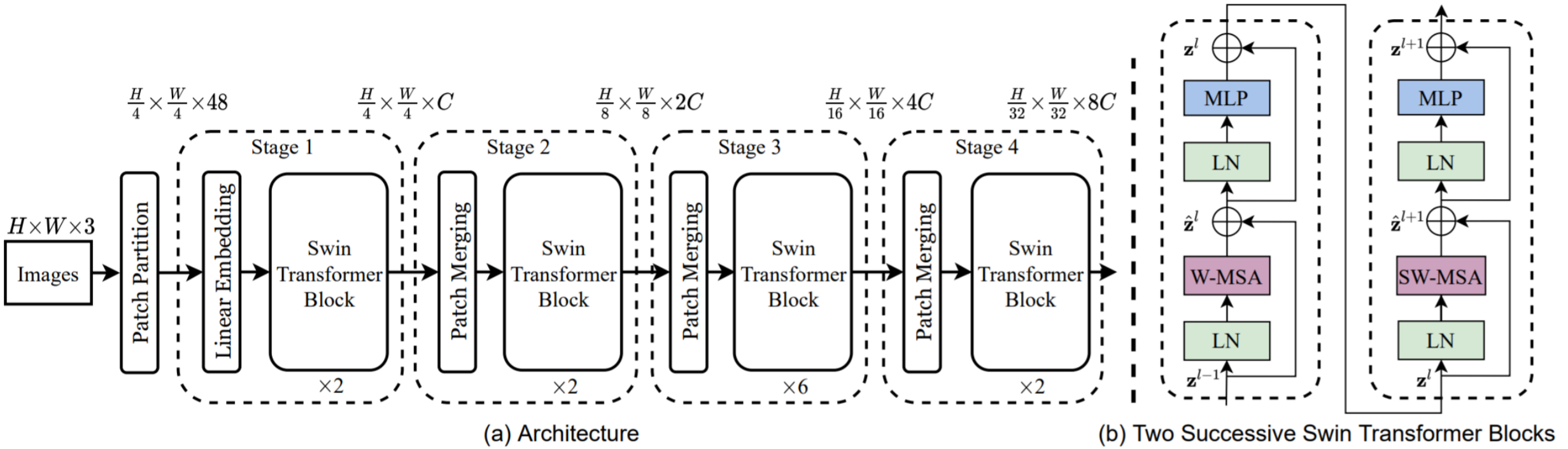


Figure 3. (a) The architecture of a Swin Transformer (Swin-T); (b) two successive Swin Transformer Blocks (notation presented with Eq. (3)). W-MSA and SW-MSA are multi-head self attention modules with regular and shifted windowing configurations, respectively.

Swin Transformer: Hierarchical Vision Transformer using Shifted Windows, ICCV 2021

Swin Transformer

- Shifted Window Attention

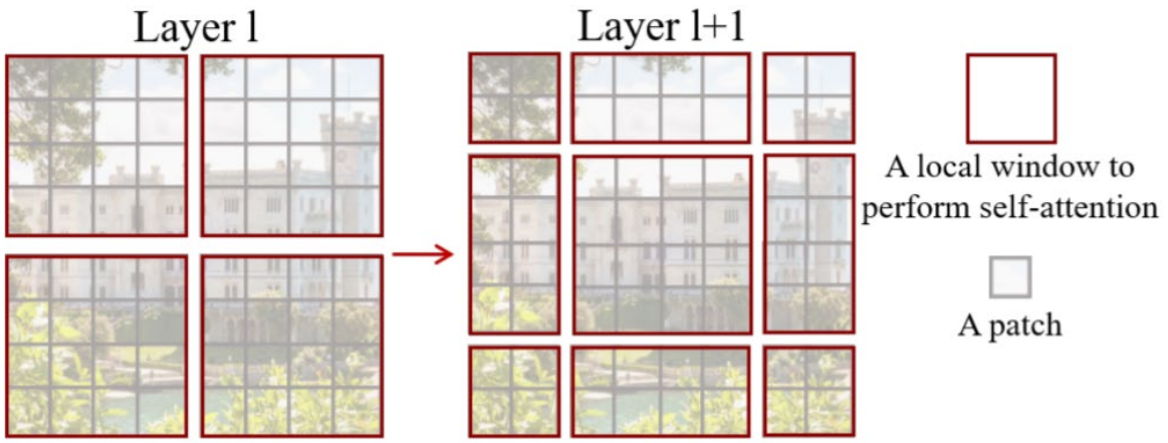


Figure 2. An illustration of the *shifted window* approach for computing self-attention in the proposed Swin Transformer architecture. In layer l (left), a regular window partitioning scheme is adopted, and self-attention is computed within each window. In the next layer $l + 1$ (right), the window partitioning is shifted, resulting in new windows. The self-attention computation in the new windows crosses the boundaries of the previous windows in layer l, providing connections among them.

Vision Transformers -> CNN

- Large Kernel
- Layer Norm
- Nonlinear Activations: GELU
- Optimization: AdamW、LAMB

- Make CNNs compete favorably with Transformers in terms of accuracy and scalability

A ConvNet for the 2020s, Arxiv 2022 (Facebook)

Scaling Up Your Kernels to 31x31: Revisiting Large Kernel Design in CNNs , Arxiv 2022 (旷视)

ConvNeXt: Training Techniques

- Begin with ResNet-50/200
- 90 -> 300 epochs
- AdamW
- Data augmentation
- Regularization schemes

- 76.1% -> 78.8% (ResNet-50)

(pre-)training config	ConvNeXt-T/S/B/L ImageNet-1K 224^2	ConvNeXt-B/L/XL ImageNet-22K 224^2
optimizer	AdamW	AdamW
base learning rate	4e-3	4e-3
weight decay	0.05	0.05
optimizer momentum	β_1, β_2=0.9, 0.999	β_1, β_2=0.9, 0.999
batch size	4096	4096
training epochs	300	90
learning rate schedule	cosine decay	cosine decay
warmup epochs	20	5
warmup schedule	linear	linear
layer-wise lr decay [6, 10]	None	None
randaugment [12]	(9, 0.5)	(9, 0.5)
label smoothing [65]	0.1	0.1
mixup [85]	0.8	0.8
cutmix [84]	1.0	1.0
stochastic depth [34]	0.1/0.4/0.5/0.5	0.1/0.1/0.2
layer scale [69]	1e-6	1e-6
gradient clip	None	None
exp. mov. avg. (EMA) [48]	0.9999	None

A ConvNet for the 2020s, Arxiv 2022

ConvNeXt: Macro Design

- Number of blocks
  - (3, 4, 6, 3) -> (3, 3, 9, s3) (ResNet-50)
  - 78.8% -> 79.4%
- Patchify
  - 7x7 convolution with stride 2 following by max pooling -> 4x4, stride 4 convolution
  - 79.4% -> 79.5%

ConvNeXt: ResNeXtify and Inverted Bottleneck

- ResNeXtify
  - Depthwise conv
  - Increase the network width
  - -> 80.5%

- Inverted Bottleneck
  - 80.5% to 80.6%

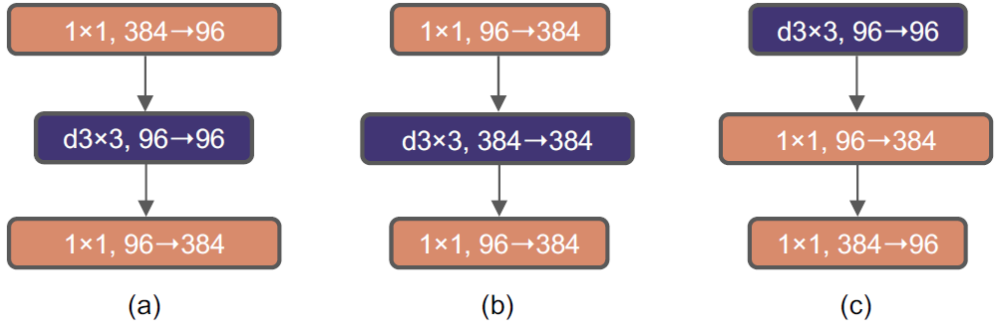
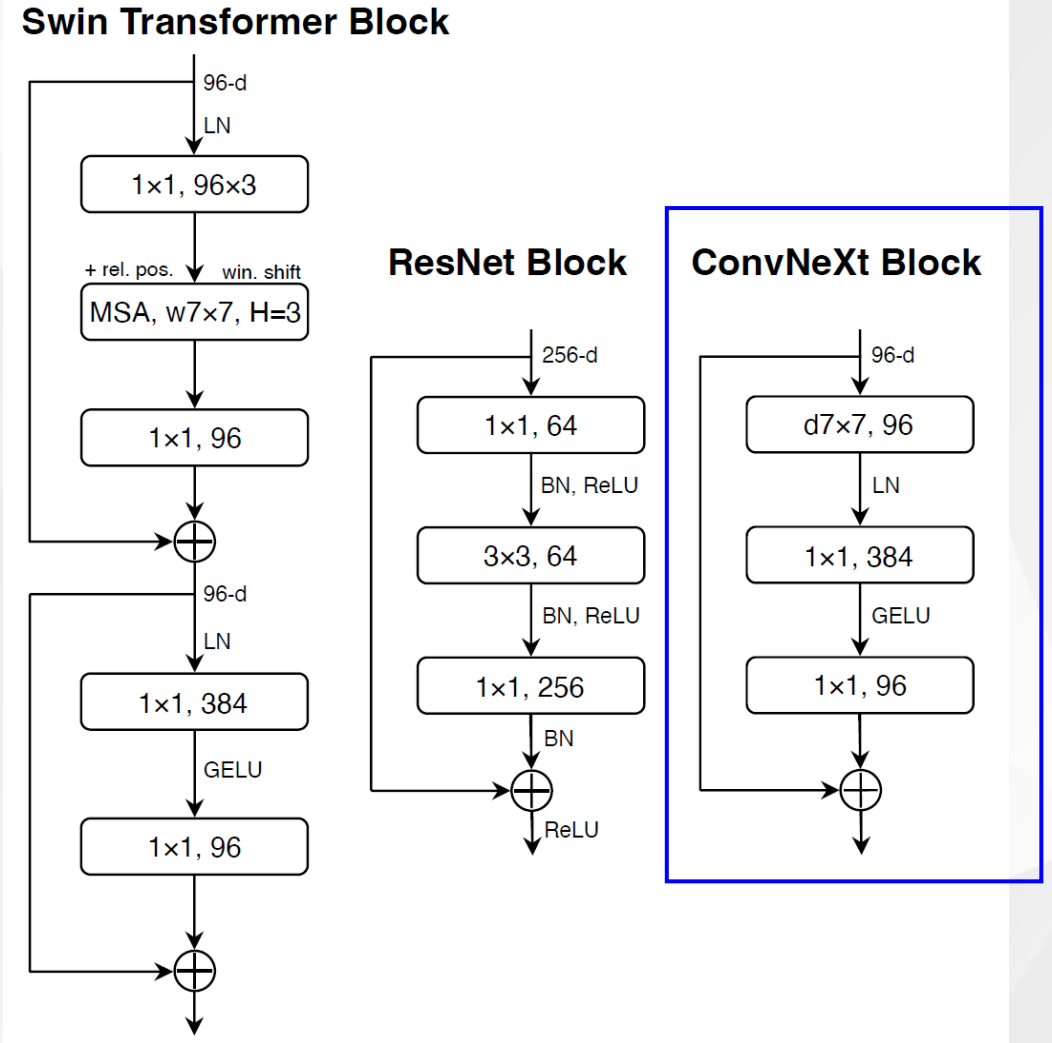


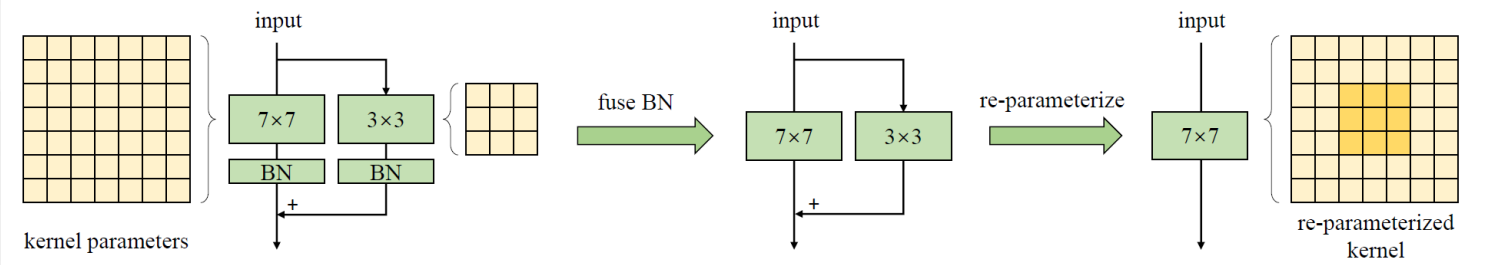
Figure 3. **Block modifications and resulted specifications. (a)** is a ResNeXt block; in **(b)** we create an inverted bottleneck block and in **(c)** the position of the spatial depthwise conv layer is moved up.

ConvNeXt: Large Kernel Sizes and Micro Design

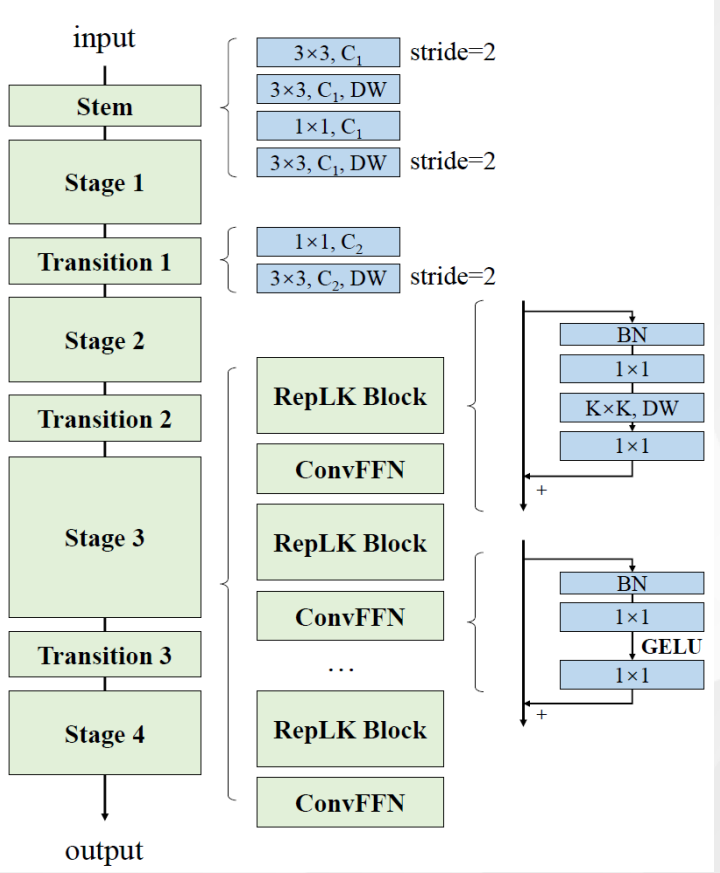
- Large Kernel Sizes
  - 3x3 -> 7x7
  - -> 80.6%
- Micro Design
  - Replacing ReLU with GELU: 80.6%
  - A single GELU activation in each block: 81.3%
  - Remove two BN layers, leaving only one BN layer before 1 x 1 conv layers: 81.4%
  - Substituting BN with LN: 81.5%
  - Separate downsampling layers: 82.0%
  - Optimization: AdamW



RepLKNet



- Re-parameterized large depth-wise convolutions
  - large depth-wise convolutions
  - identity shortcut is vital
  - re-parameterizing
  - large convolutions boost downstream tasks much more
  - large kernel is useful even on small feature maps



Scaling Up Your Kernels to 31x31: Revisiting Large Kernel Design in CNNs, Arxiv 2022

Summary

- 掌握近年来的主流卷积和Transformer网络结构
- 结合实际问题和任务灵活使用
- 持续了解和跟进网络架构研究进展
- 自己能有一些思考、拓展和突破

Representative Vision Tasks and Deep Learning Models

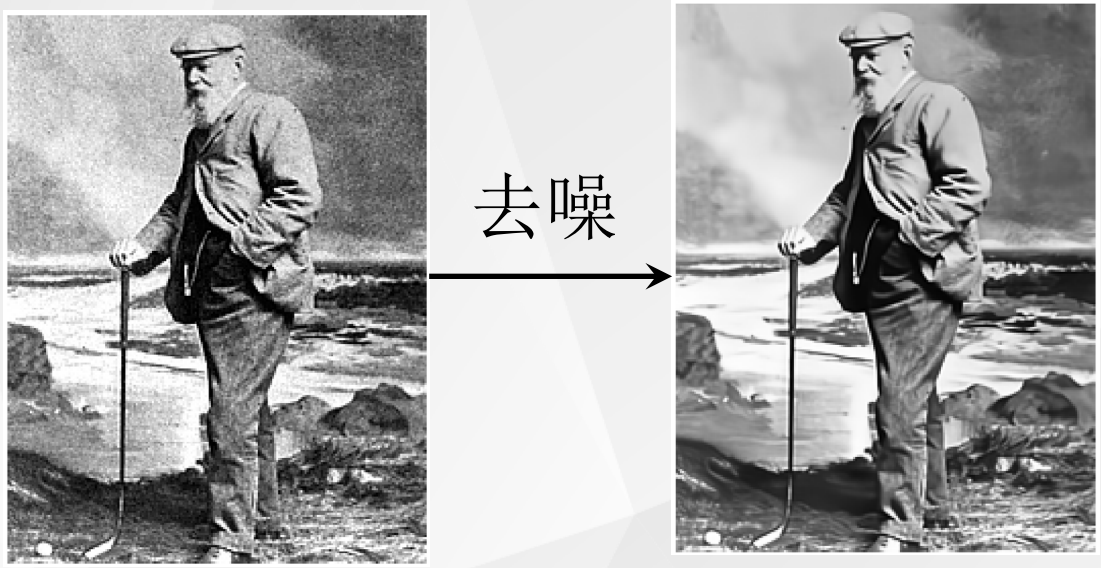
Wangmeng Zuo

Centre on Machine Learning Research,
Harbin Institute of Technology

Content

- Introduction
- Low Level Vision: Image Restoration and Generation
- High Level Vision: Visual Understanding
- Vision and Language

典型视觉学习任务（底层视觉）

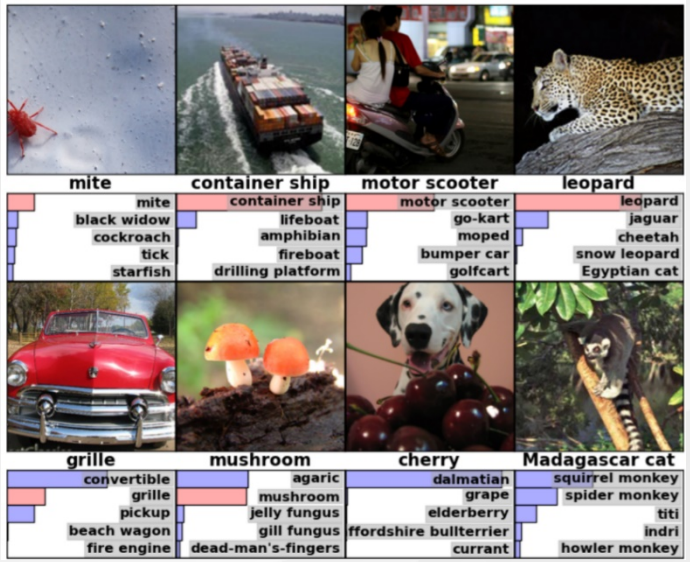


风格迁移

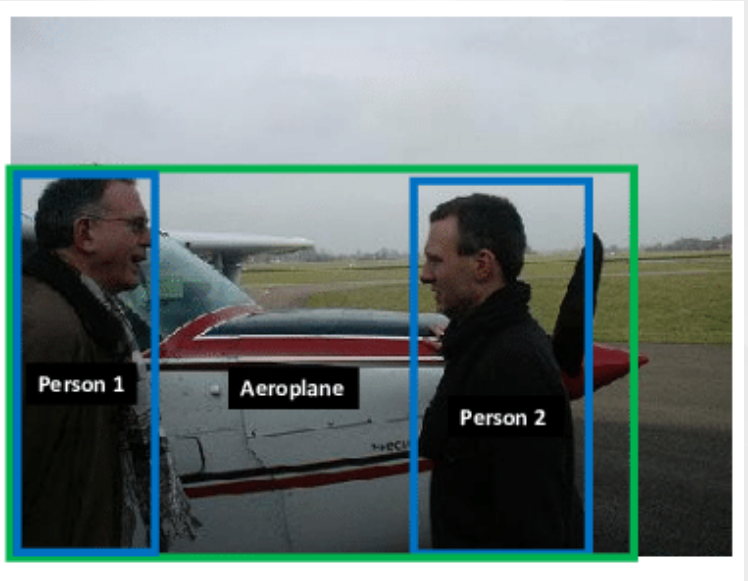
修复

典型视觉学习任务（视觉理解）

图像级分类



边界框级检测



物体关系预测

像素级分割

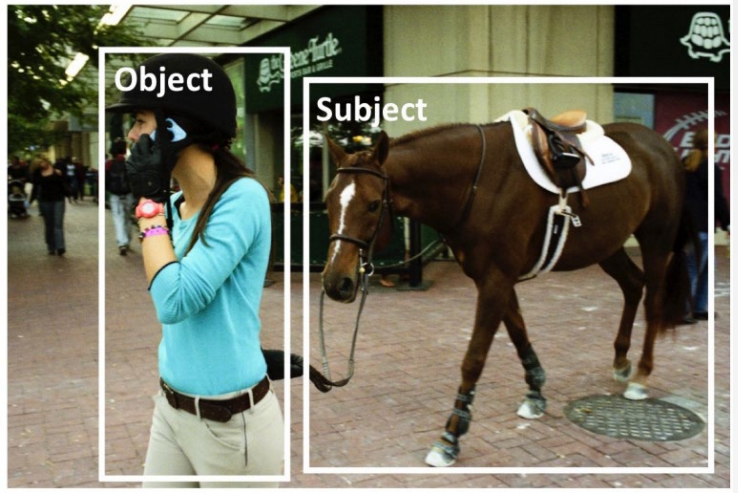
典型视觉任务（视觉-自然语言）

自然语言描述

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."



指代消解

视觉问答

Q: Does this foundation have any sunscreen?
A: yes

Q: What is this?
A: 10 euros

Q: What color is this?
A: green

Q: Please can you tell me what this item is?
A: butternut squash red pepper soup

Q: Is it sunny outside?
A: yes

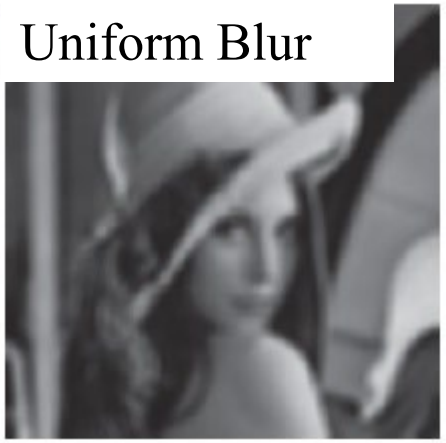
Q: Is this air conditioner on fan, dehumidifier, or air conditioning?
A: air conditioning

Content

- Introduction
- Low Level Vision: Image Restoration and Generation
  - Representative Tasks
  - Representative Networks
  - Objective Function
- High Level Vision: Visual Understanding
- Vision and Language

Image Restoration

Uniform Blur



Camera Shake

Defocus

Reflection by Glass

Noise

Motion Blur

Fog

Rain

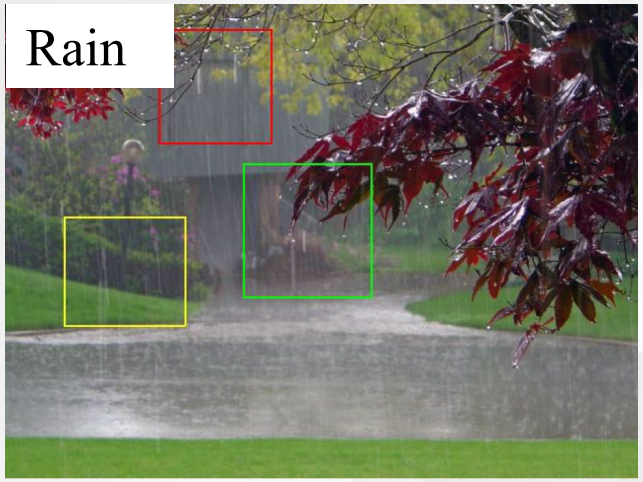
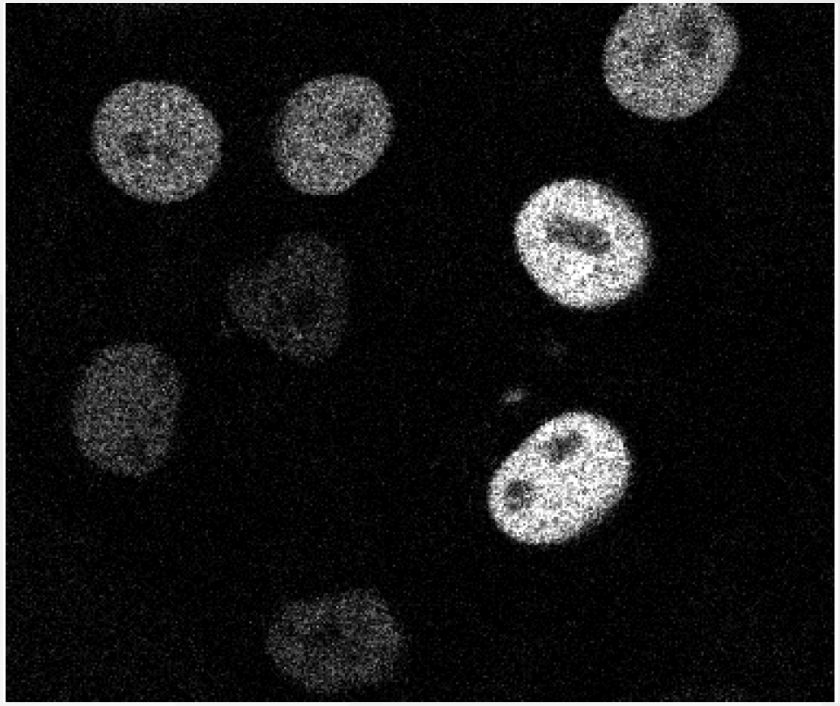


Image Restoration: Denoising

- Low-light Noise

- Microscopy



- Low-dose CT Images

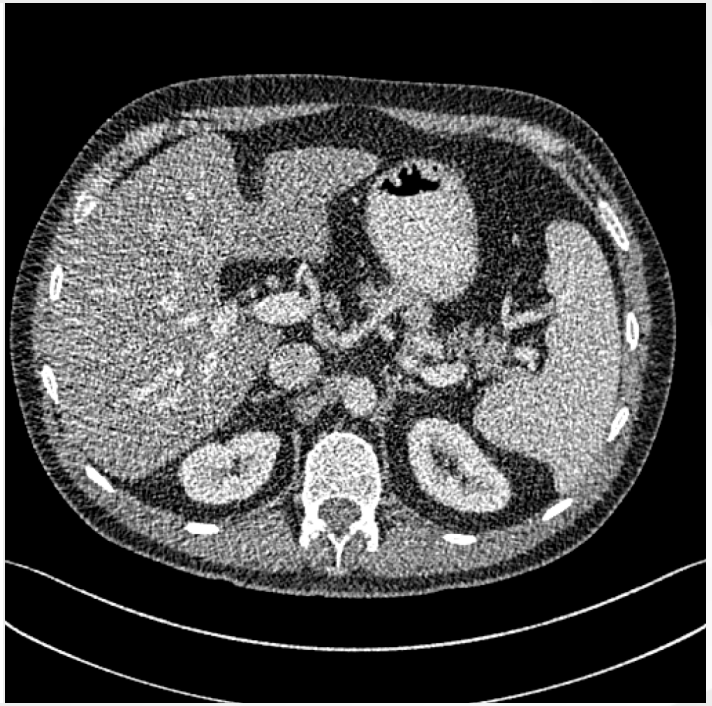


Image Super-resolution

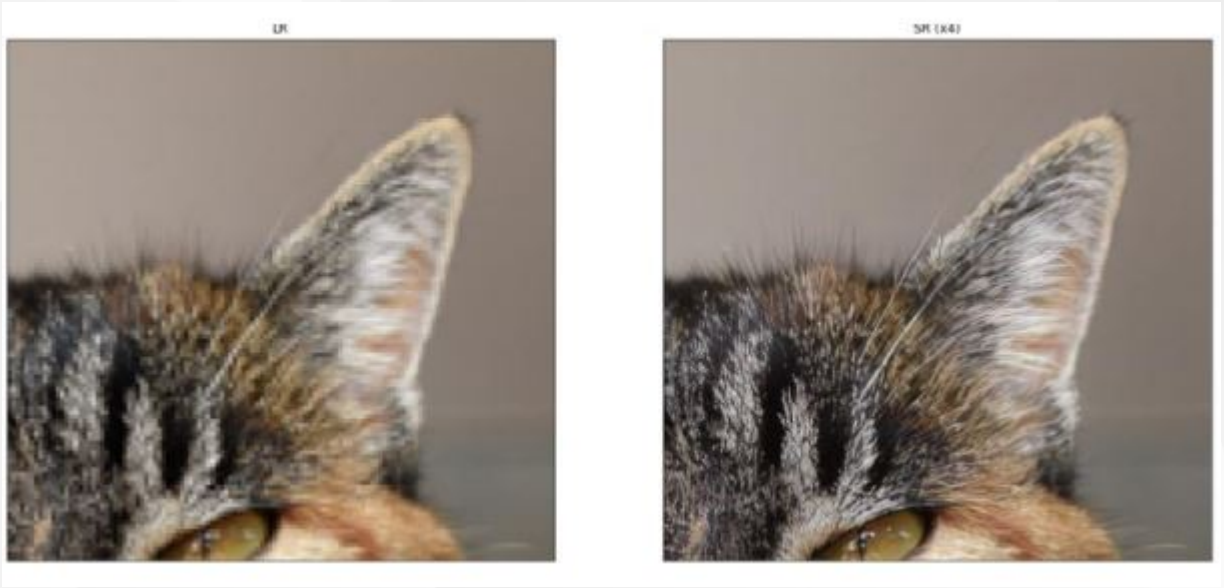
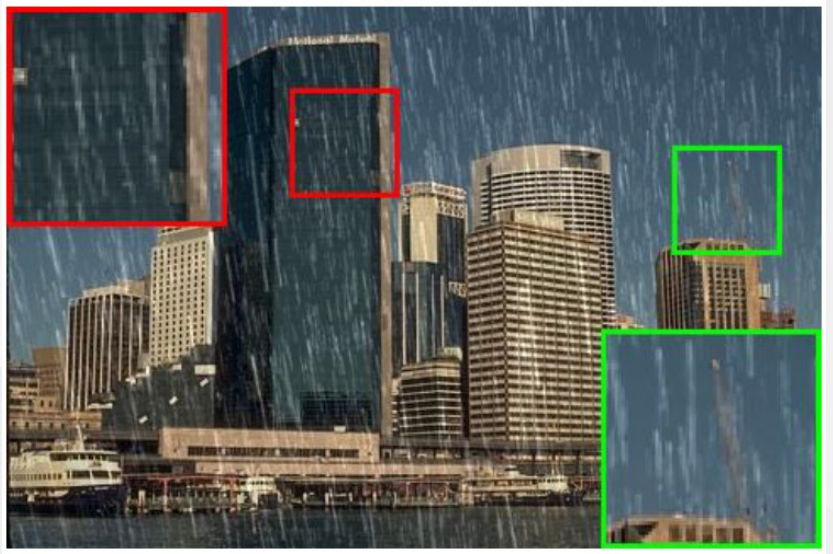
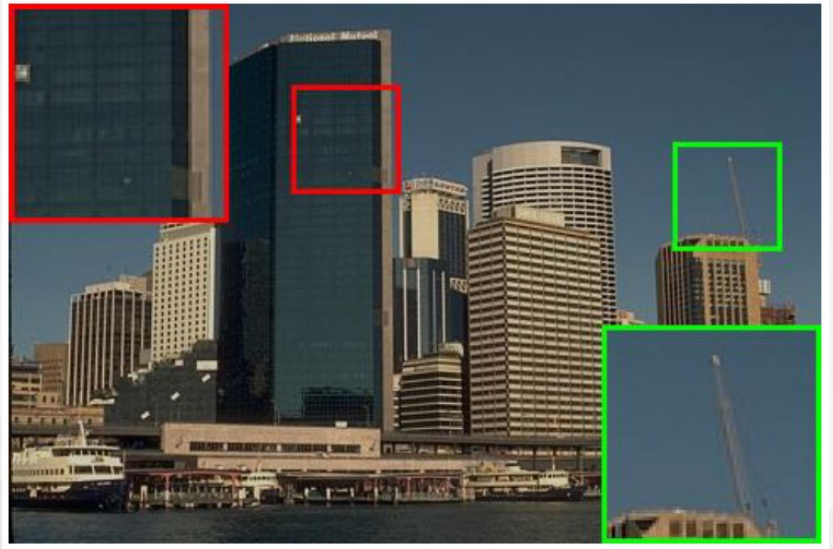


Image Deblurring

Dehazing, Deraining

Dehazing





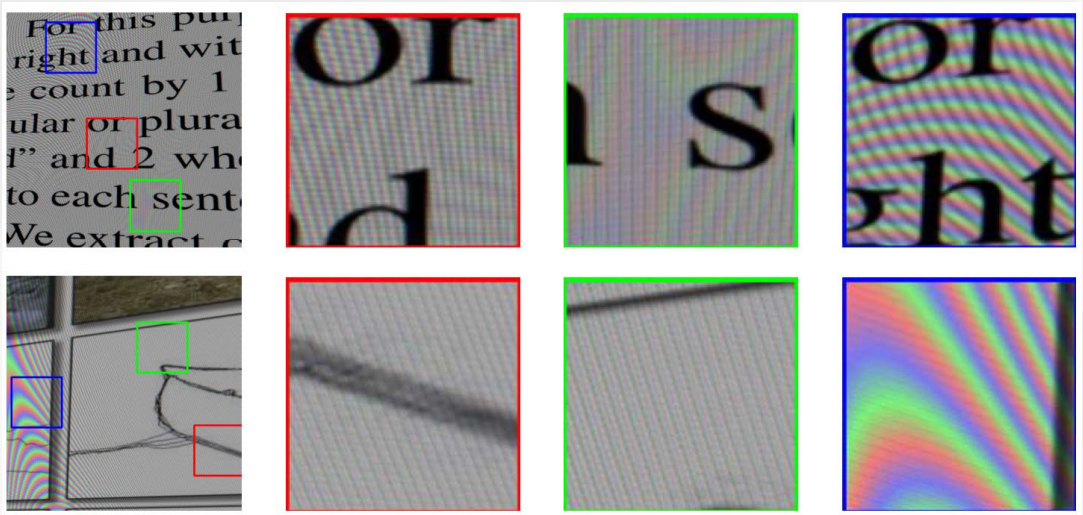
Deraining

Low Light Image Enhancement

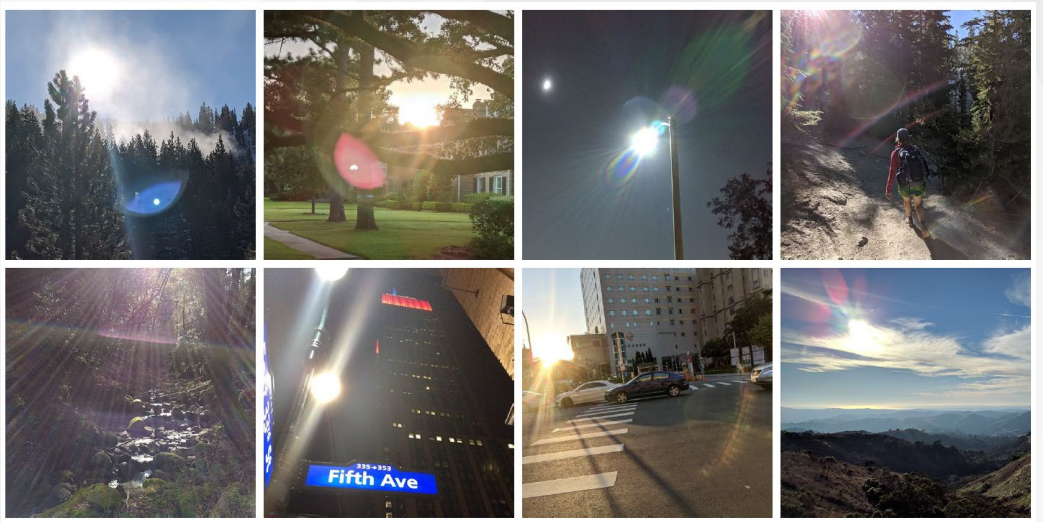
Reflection Removal

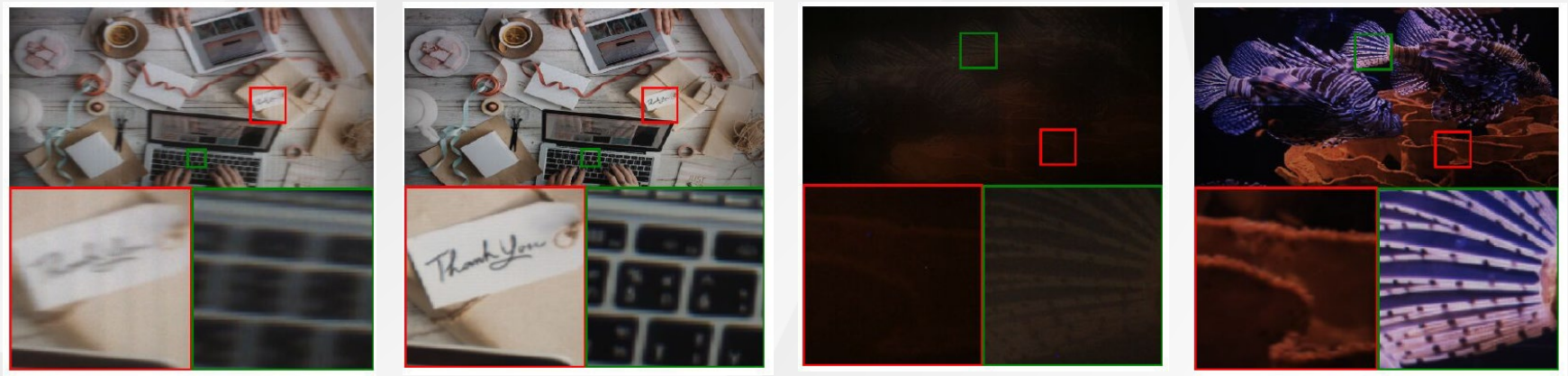
Others

Demoireing



Flare Removal



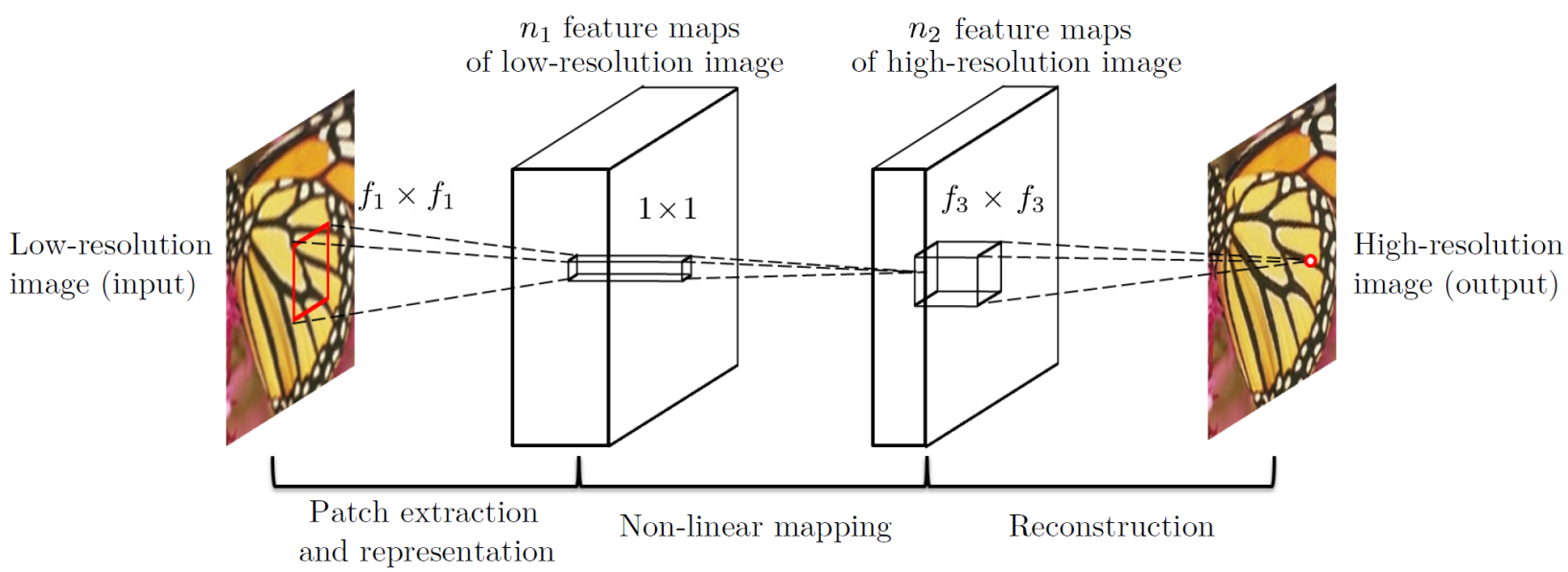


Under-Display Camera: T-OLED & P-OLED

Content

- Introduction
- Low Level Vision: Image Restoration and Generation
  - Representative Tasks
  - Representative Networks
  - Objective Function
- High Level Vision: Visual Understanding
- Vision and Language

Image Super-resolution: SRCNN



- Relationship to Sparse-Coding-Based Methods
  - Input: Patch
  - Convolution: Dictionary, coefficients
  - Reconstruction

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang, Learning a Deep Convolutional Network for Image Super-Resolution, ECCV 2016.

Image Super-resolution: RCAN

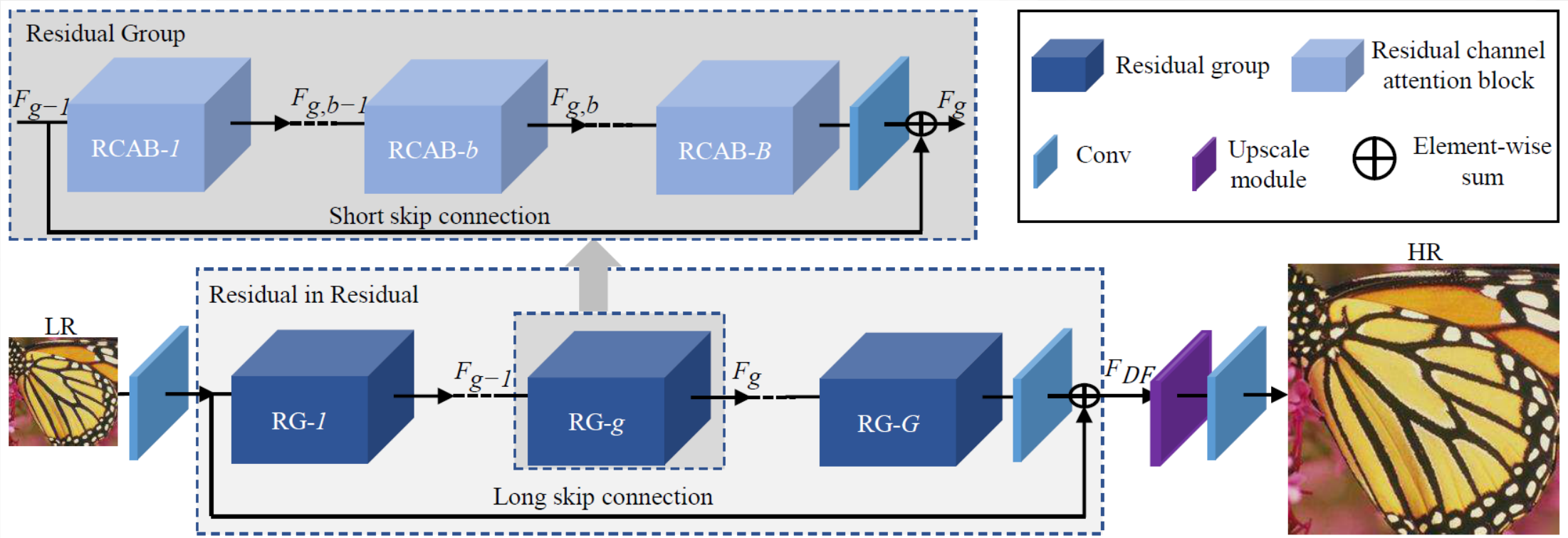
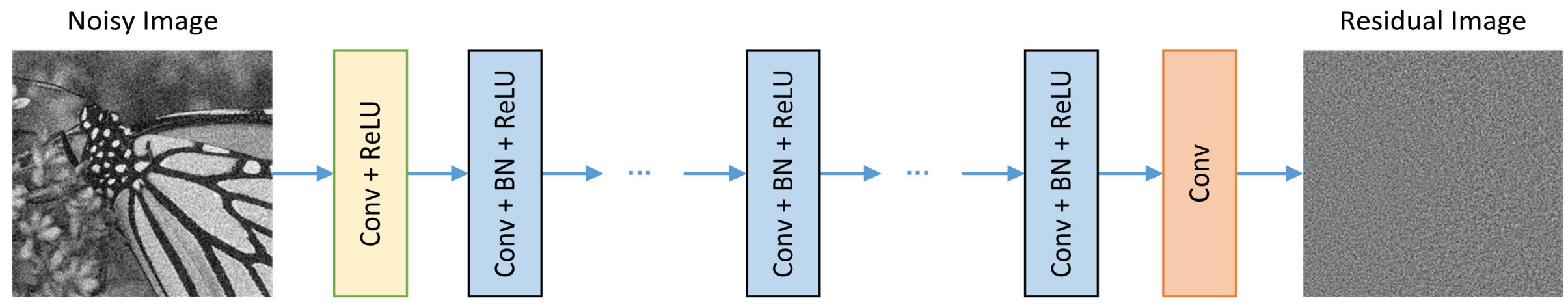


Image Super-Resolution Using Very Deep Residual Channel Attention Networks, ECCV 2018.

Image Denoising: DnCNN



- Officially included in Matlab since Matlab 2017b
  - https://www.mathworks.com/help/images/ref/denoisingnetwork.html

Description

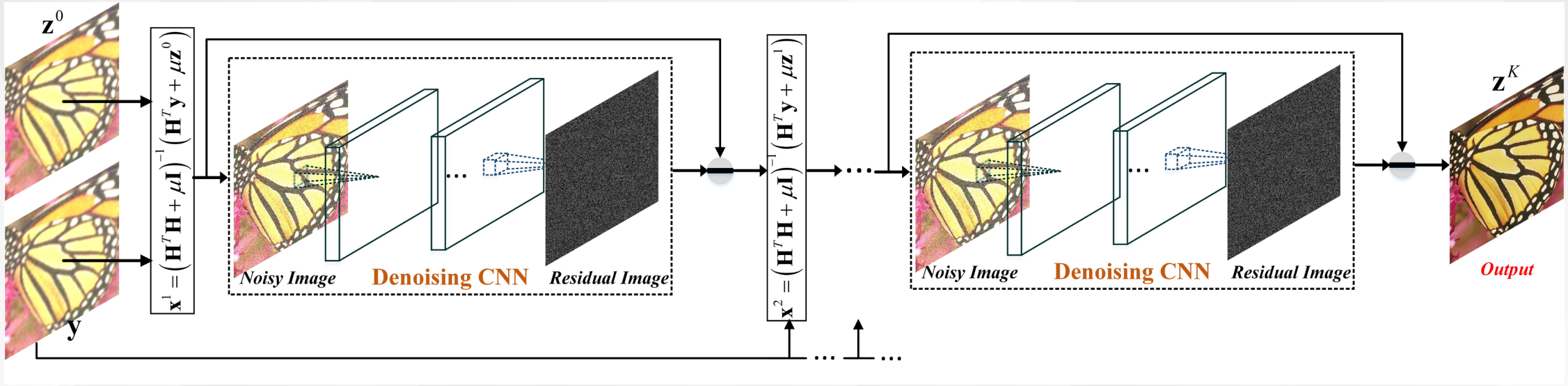
`net = denoisingNetwork(modelName)` returns a pretrained image denoising deep neural network specified by `modelName`.

This function requires that you have Deep Learning Toolbox™.

References

[1] Zhang, K., W. Zuo, Y. Chen, D. Meng, and L. Zhang. "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising." *IEEE Transactions on Image Processing*. Vol. 26, Number 7, Feb. 2017, pp. 3142-3155.

Image Deblurring: IRCNN



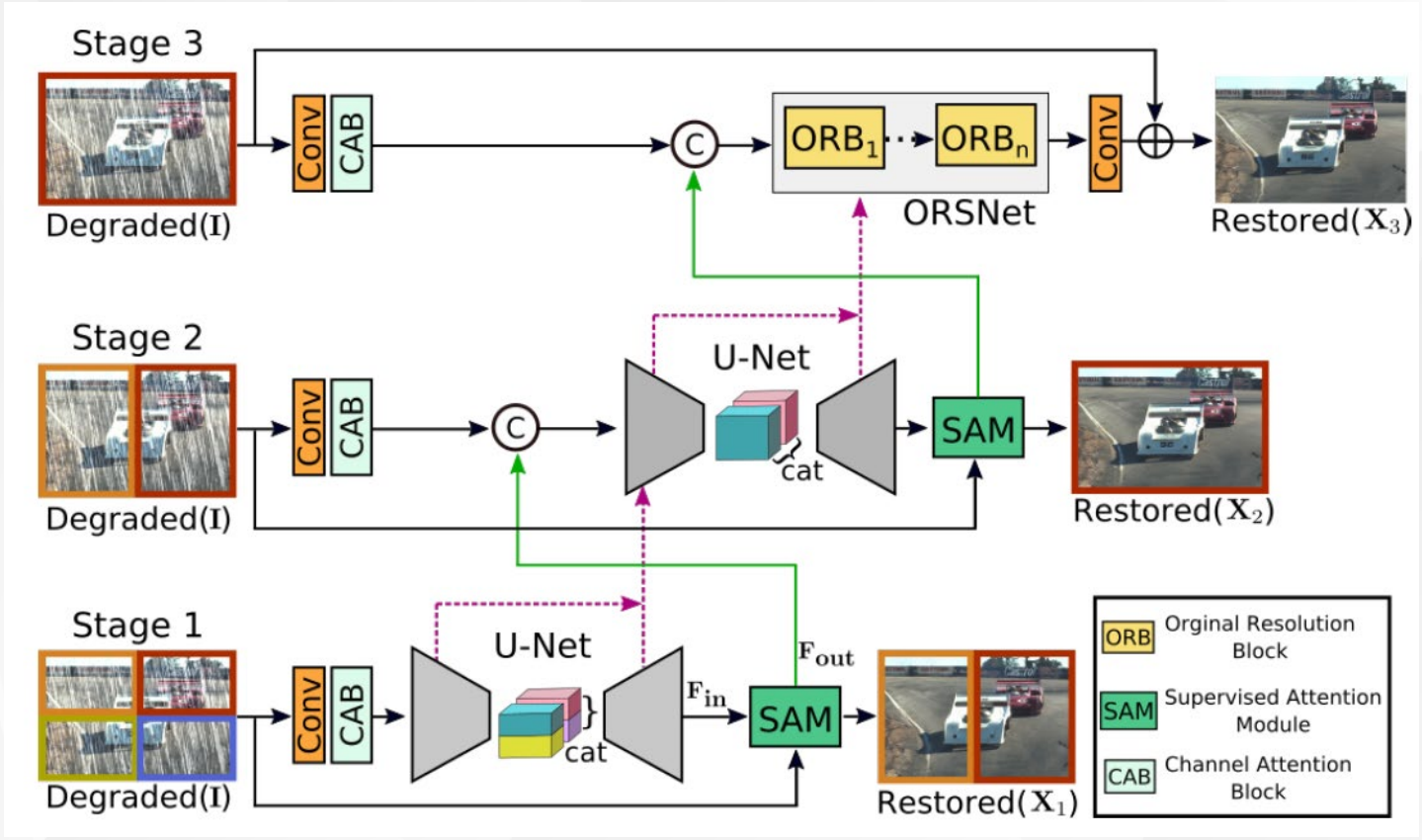
$$\mathcal{L}_\mu(\mathbf{x}, \mathbf{z}) = \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \lambda\Phi(\mathbf{z}) + \frac{\mu}{2}\|\mathbf{z} - \mathbf{x}\|^2$$

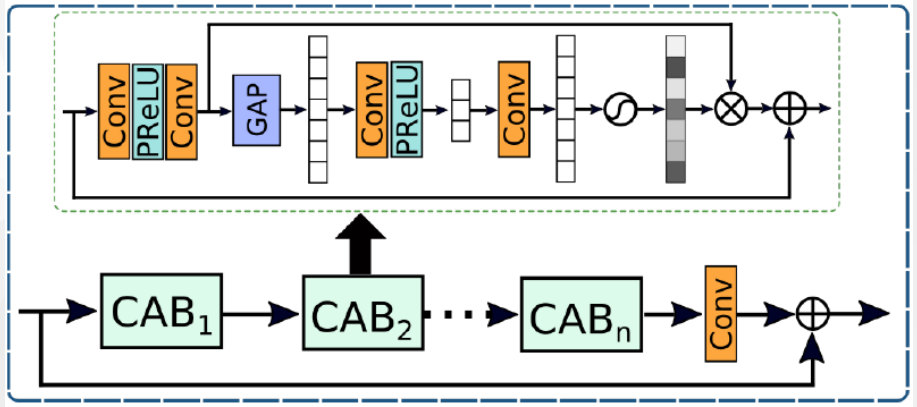
$$\begin{cases} \mathbf{x}_{k+1} = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \mu\|\mathbf{x} - \mathbf{z}_k\|^2 \\ \mathbf{z}_{k+1} = \arg\min_{\mathbf{z}} \frac{\mu}{2}\|\mathbf{z} - \mathbf{x}_{k+1}\|^2 + \lambda\Phi(\mathbf{z}) \end{cases}$$

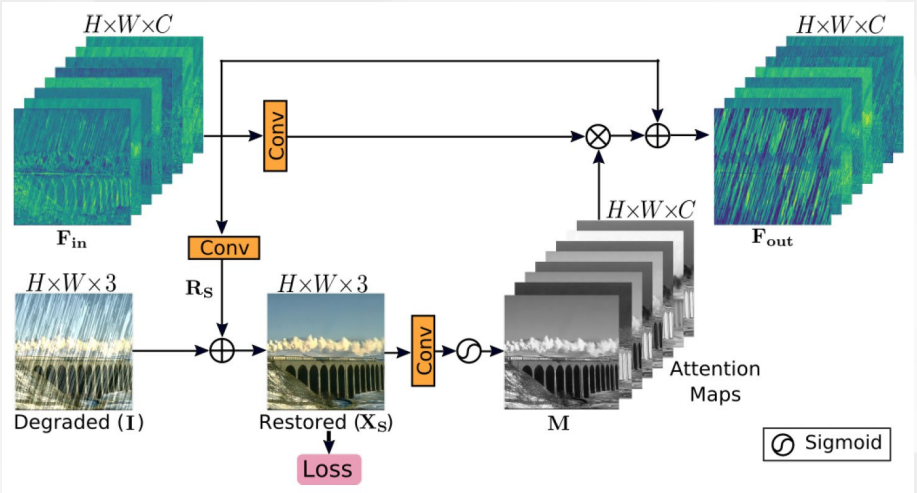
$$\mathbf{z}_{k+1} = Denoiser(\mathbf{x}_{k+1}, \sqrt{\lambda/\mu})$$

Learning Deep CNN Denoiser Prior for Image Restoration, CVPR 2017

Image Deblurring: MPRNet

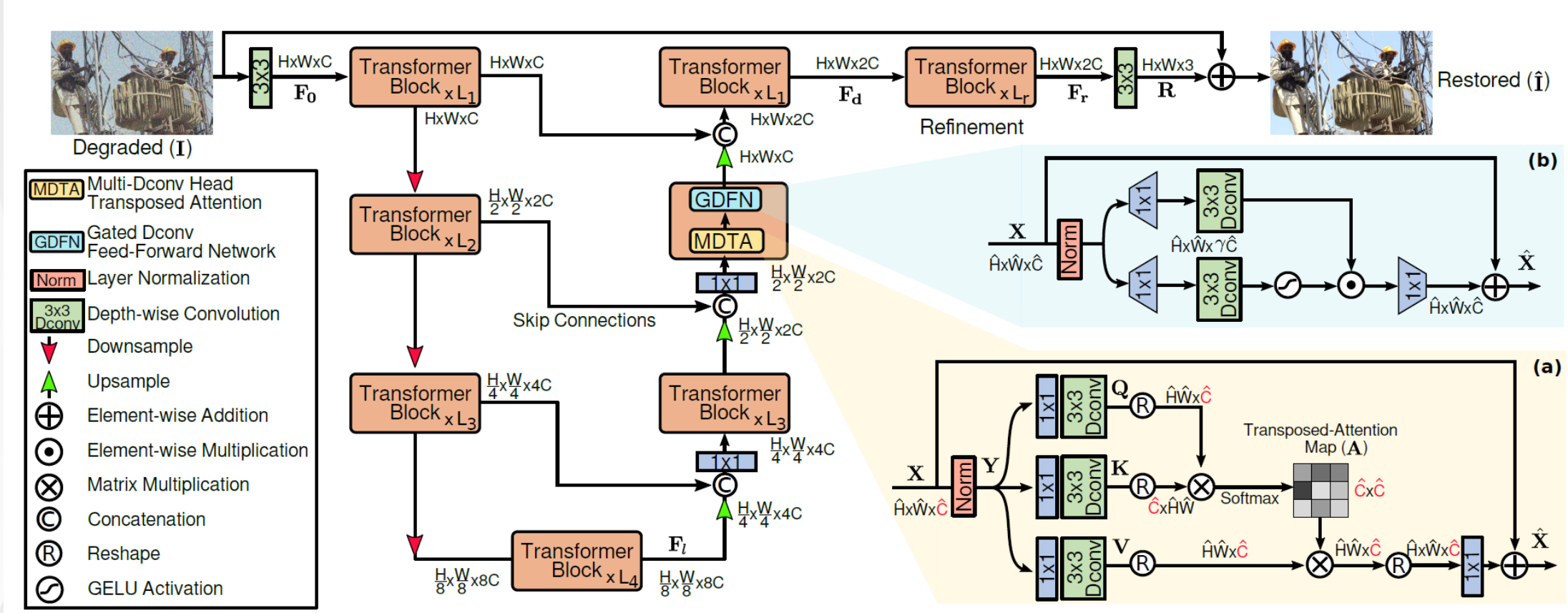






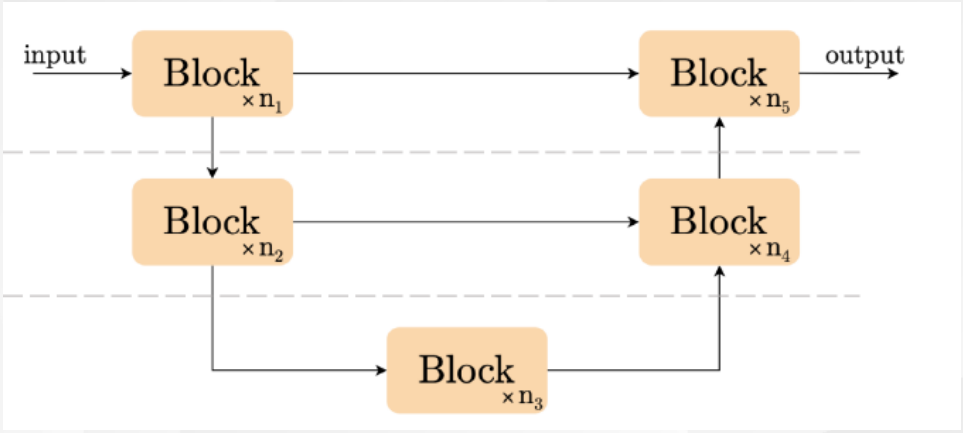
Multi-Stage Progressive Image Restoration, CVPR 2021.

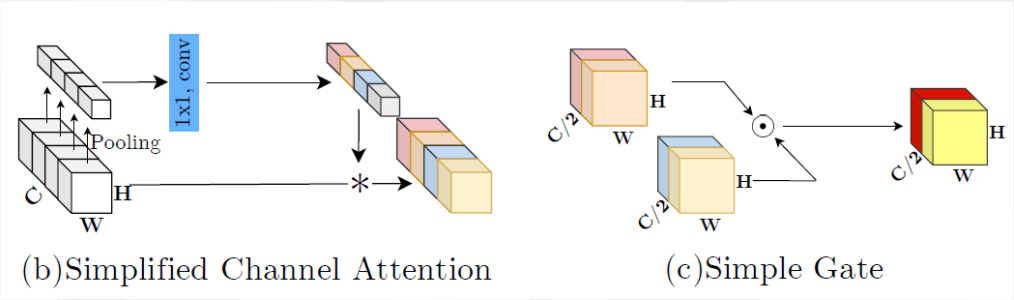
Restormer



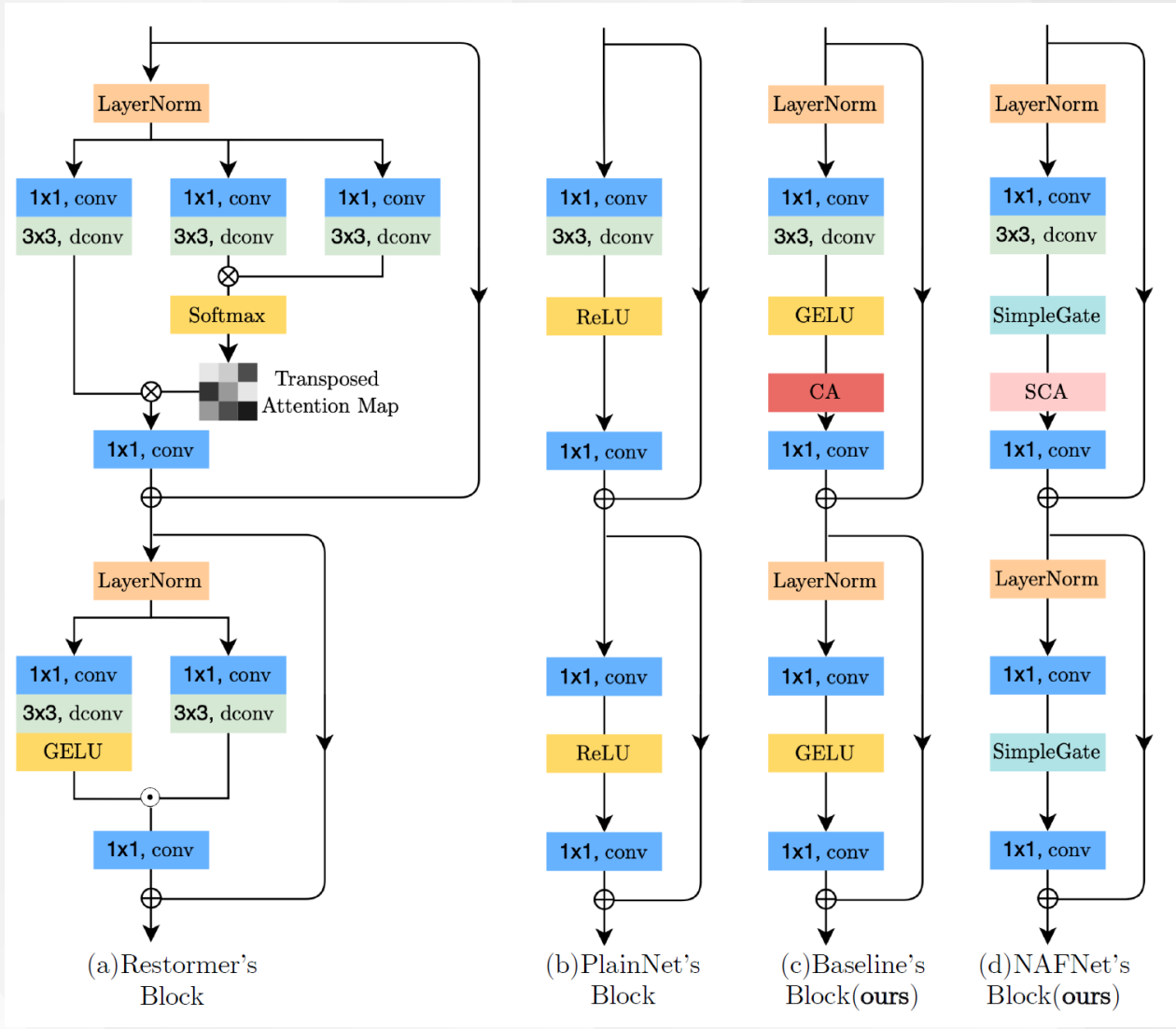
Restormer: Efficient Transformer for High-Resolution Image Restoration, CVPR2022

NAFNet





(b)Simplified Channel Attention (c)Simple Gate



(a)Restormer's Block (b)PlainNet's Block (c)Baseline's Block(**ours**) (d)NAFNet's Block(**ours**)

Simple Baselines for Image Restoration, ECCV 2022.

Content

- Introduction
- Low Level Vision: Image Restoration and Generation
  - Representative Tasks
  - Representative Networks
  - Objective Function
- High Level Vision: Visual Understanding
- Vision and Language

Objective Function

- MSE Loss

$$MSE = \frac{1}{n}\frac{1}{m}\sum_{i=1}^{n}\sum_{j=1}^{m}(Y(i,j) - \hat{Y}(i,j))^2$$

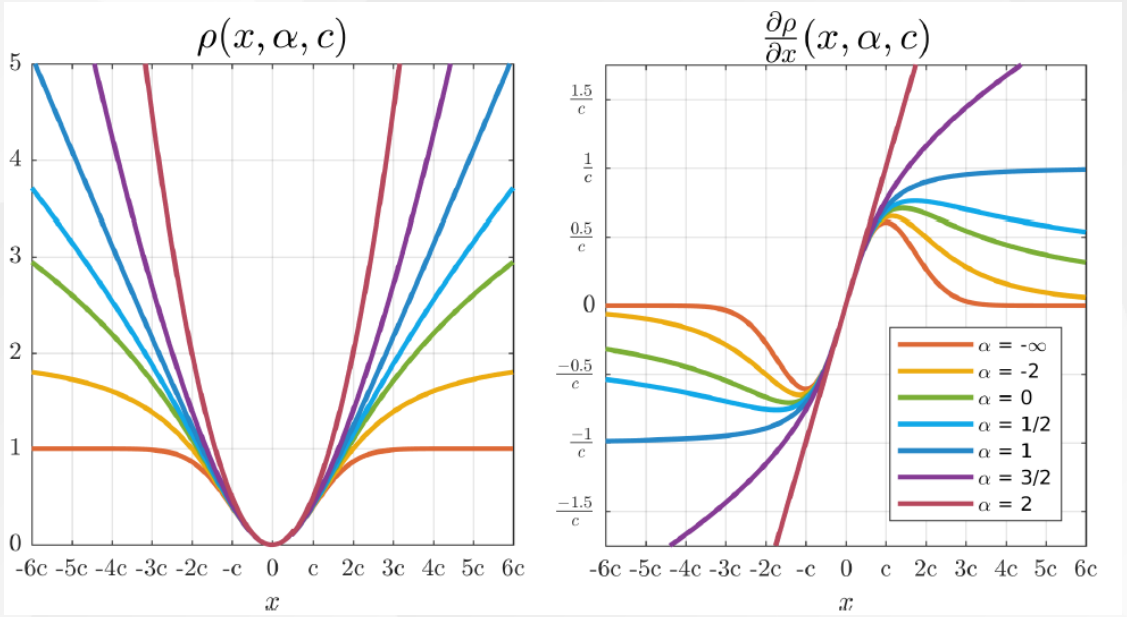
- L1-Norm Loss

$$L(\Theta) = \frac{1}{N}\sum_{i=1}^{N}\left\|H_{RCAN}\left(I_{LR}^{i}\right) - I_{HR}^{i}\right\|_1$$

Objective Function

- Generalized Charbonnier

$$\rho\left(x,\alpha,c\right)=\begin{cases}\frac{1}{2}\left(x/c\right)^2 & \text{if } \alpha=2\\ \log\left(\frac{1}{2}\left(x/c\right)^2+1\right) & \text{if } \alpha=0\\ 1-\exp\left(-\frac{1}{2}\left(x/c\right)^2\right) & \text{if } \alpha=-\infty\\ \frac{|\alpha-2|}{\alpha}\left(\left(\frac{\left(x/c\right)^2}{|\alpha-2|}+1\right)^{\alpha/2}-1\right) & \text{otherwise}\end{cases}$$



- Perceptual loss

$$\ell_{feat}^{\phi,j}(\hat{y},y)=\frac{1}{C_jH_jW_j}\|\phi_j(\hat{y})-\phi_j(y)\|_2^2$$

A General and Adaptive Robust Loss Function, ICCV 2019

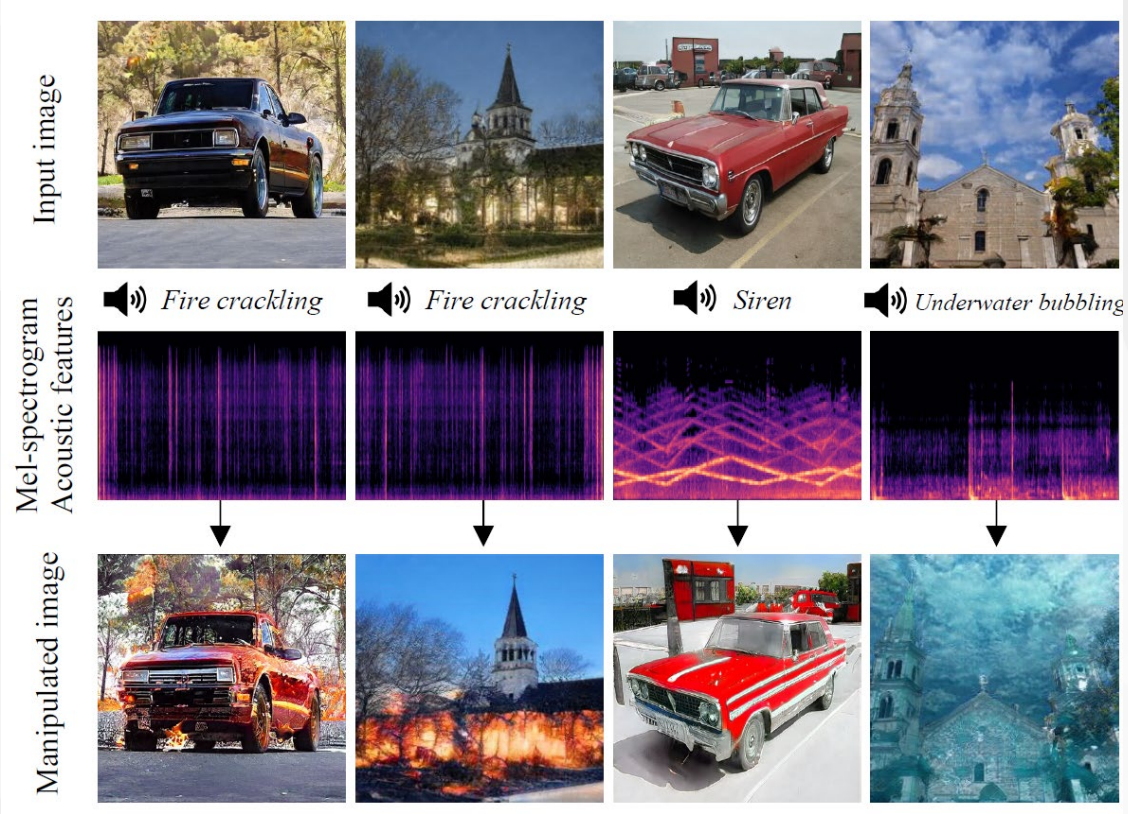
Perceptual Losses for Real-Time Style Transfer and Super-Resolution, ECCV 2016.

Some Challenging Issues

- Ill-posed problems
- Misalignment
- Learning without ground-truth

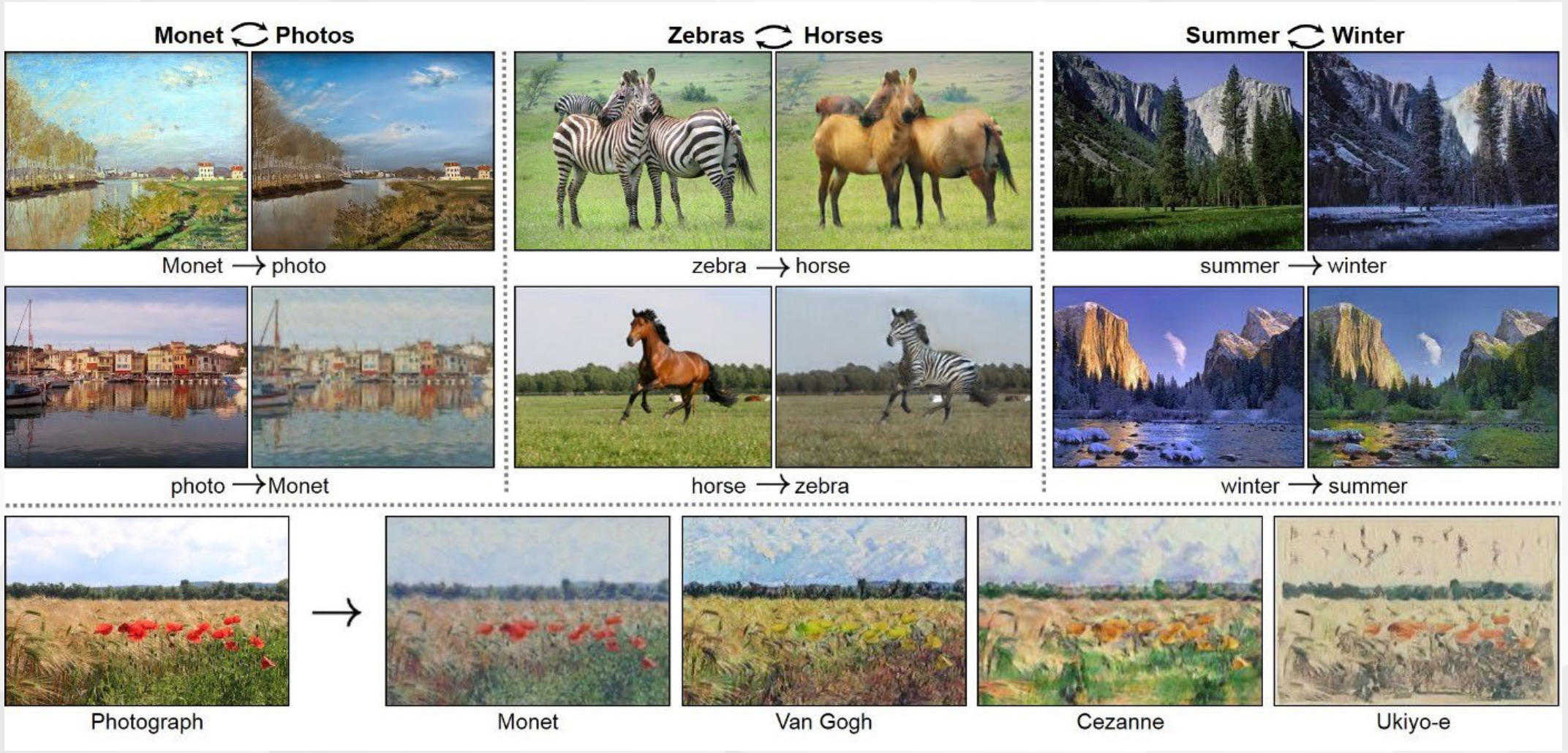
Image Generation

Image Generation



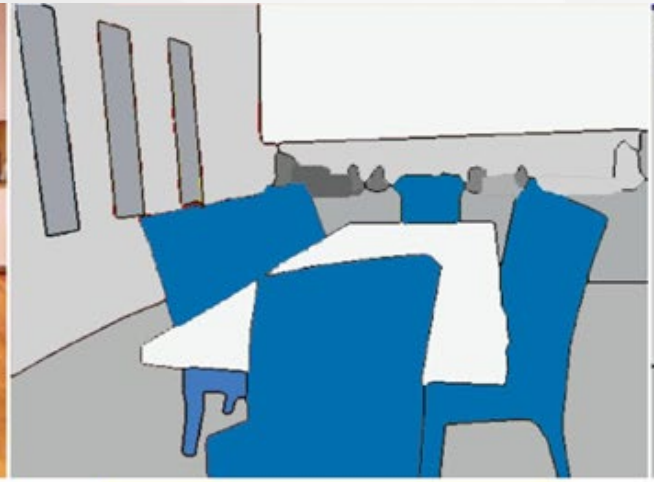
Conditional Image Generation

Image-to-Image Translation



Semantic Image Generation

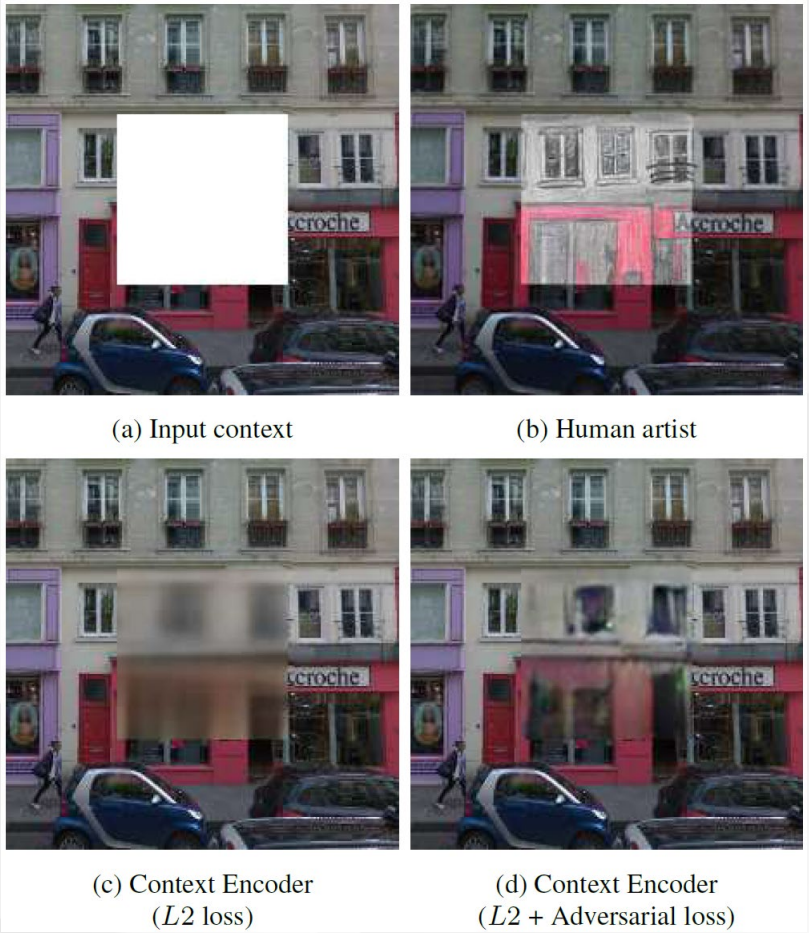
Input Image



Semantic Segmentation

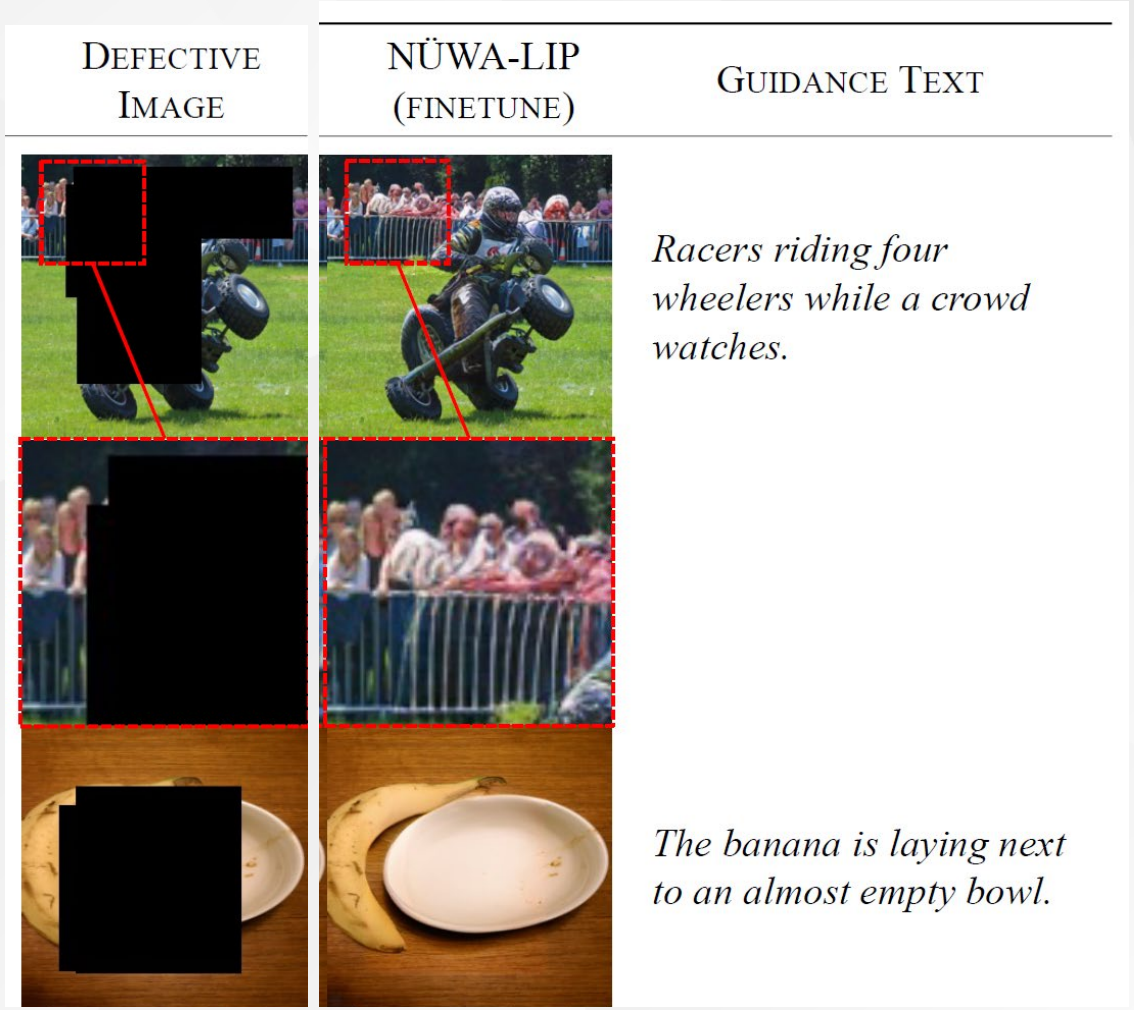
Instance Segmentation

Image Inpainting

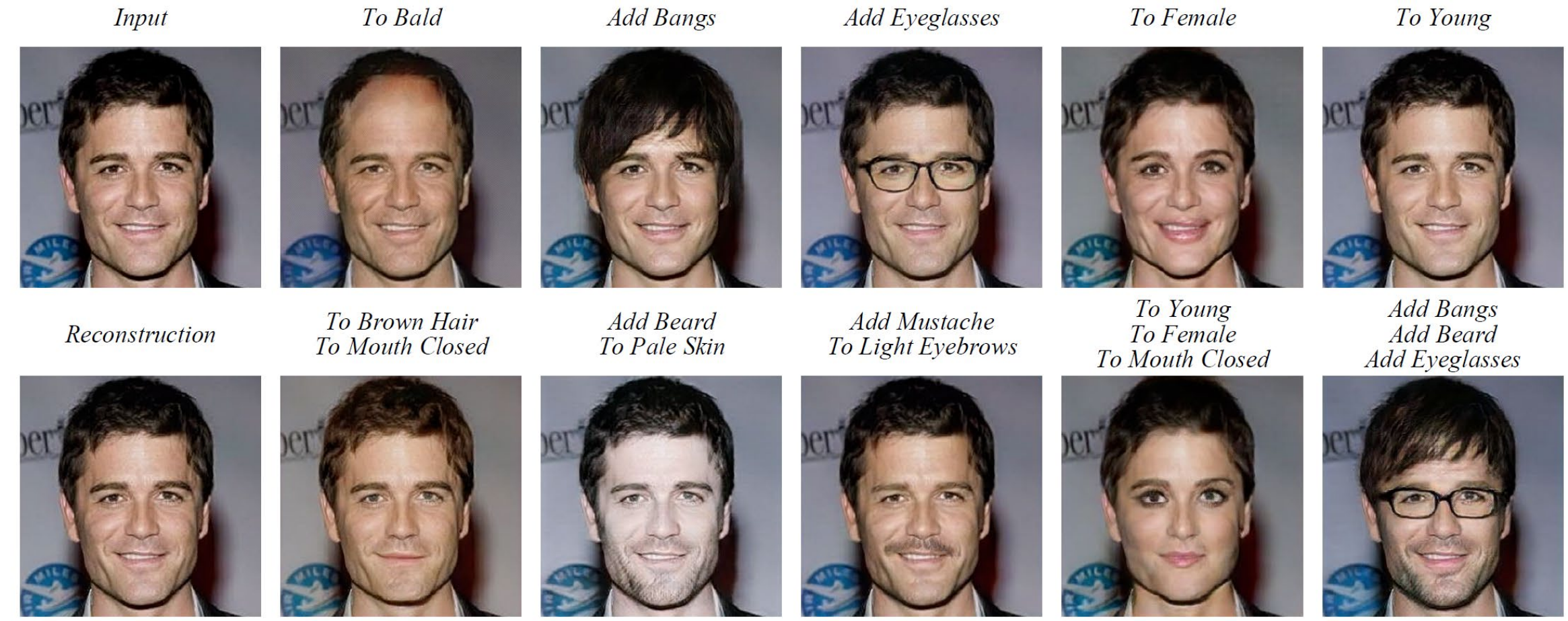


(a) Input context (b) Human artist

(c) Context Encoder ($L2$ loss) (d) Context Encoder ($L2$ + Adversarial loss)



Face Attribute Editing

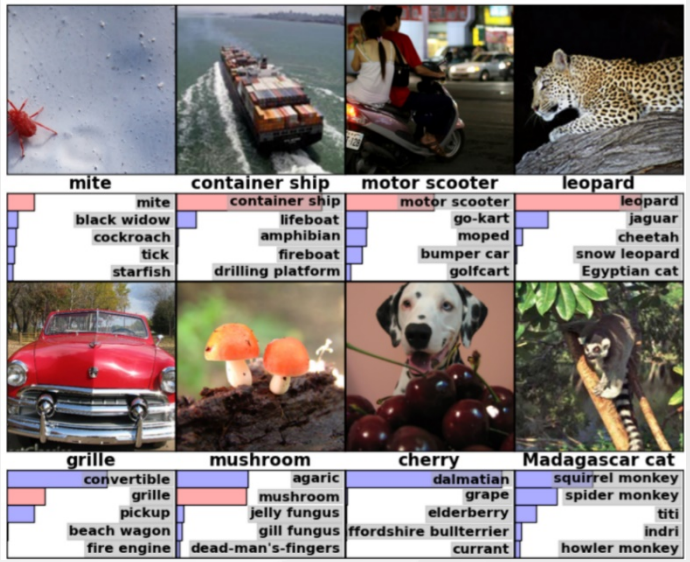


Content

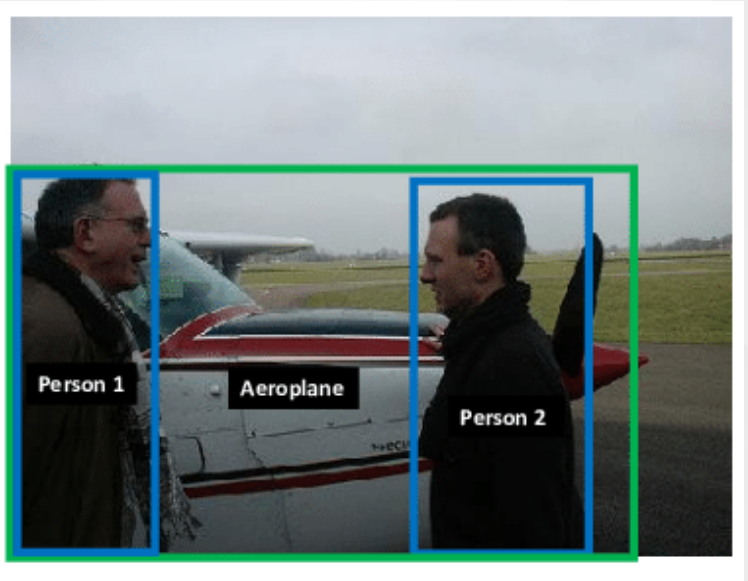
- Introduction
- Low Level Vision: Image Restoration and Generation
- High Level Vision: Visual Understanding
  - Image Classification
  - Object Detection
  - Semantic/Instance Segmentation
- Vision and Language

典型视觉学习任务（视觉理解）

图像级分类



边界框级检测



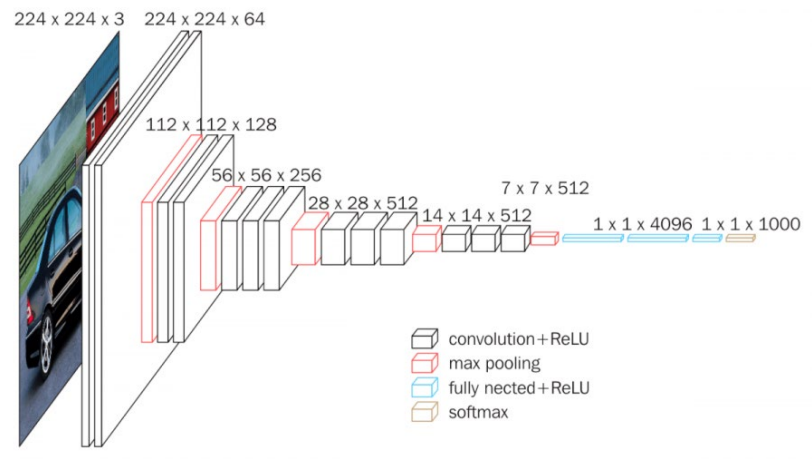
物体关系预测

像素级分割

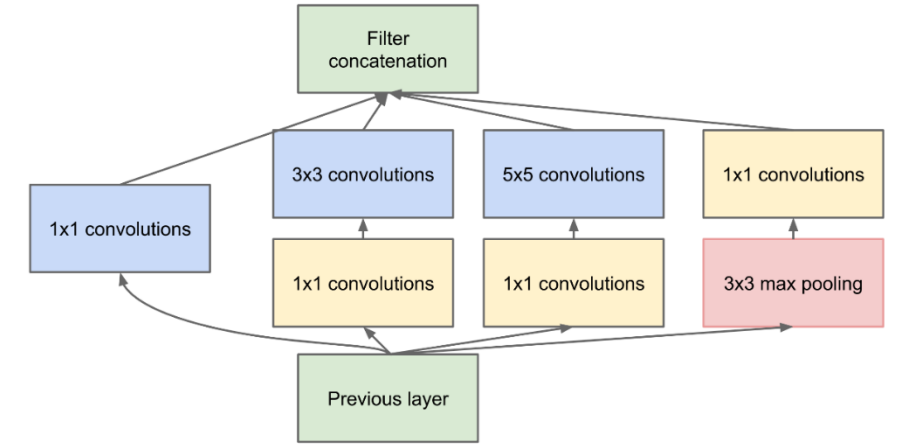
ImageNet

- 1000 object categories
- 1.2M training
- 100K testing

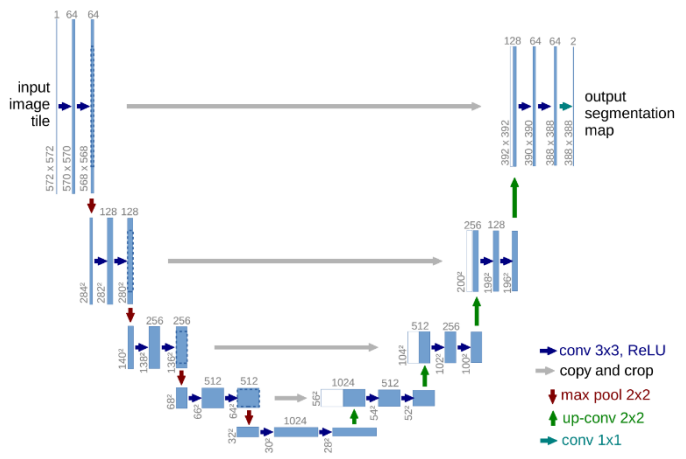
Renaissance of CNNs



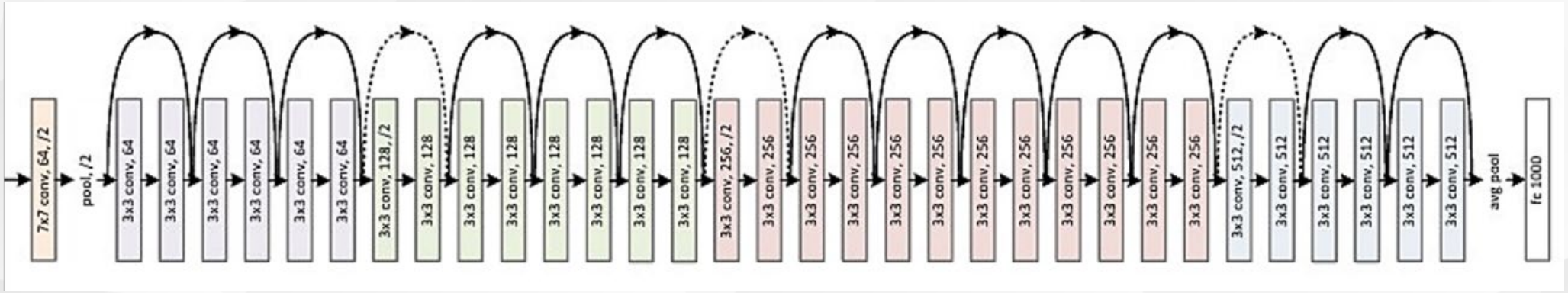
VGGNet



Inception



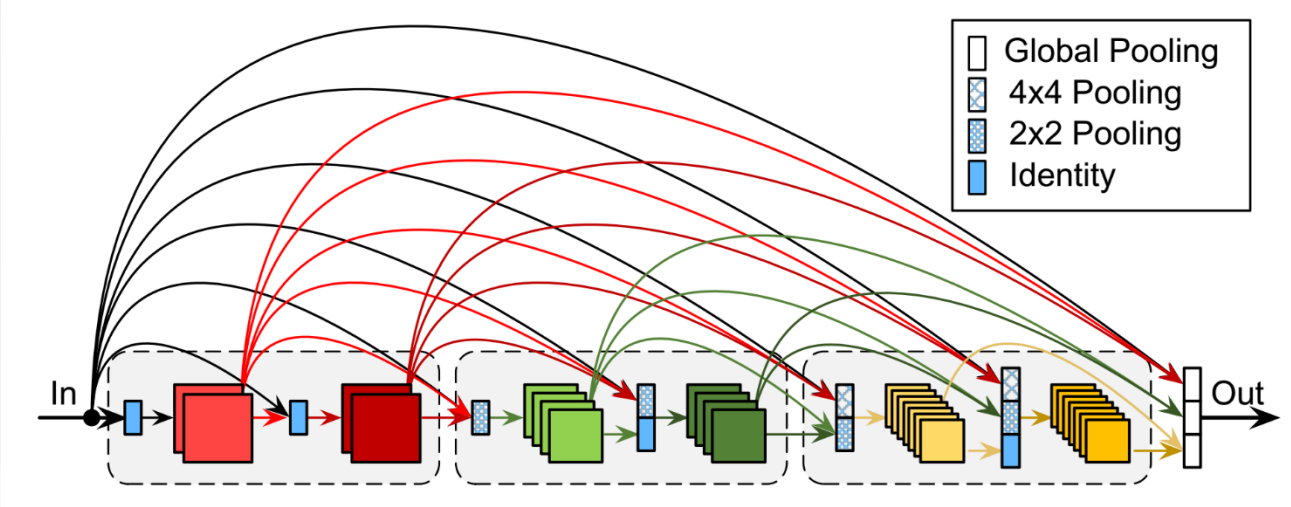
U-Net



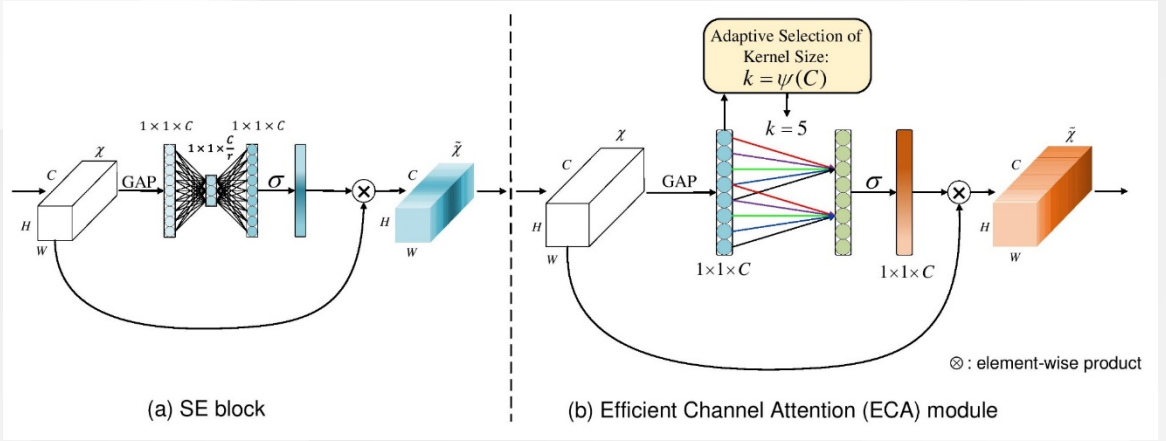
ResNet

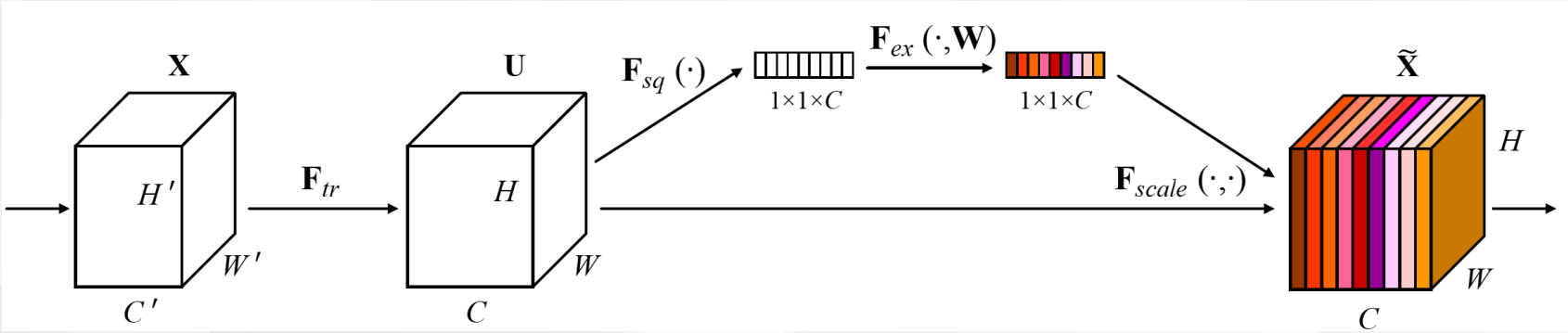
Renaissance of CNNs

DenseNet (Huang et al., CVPR 2017)



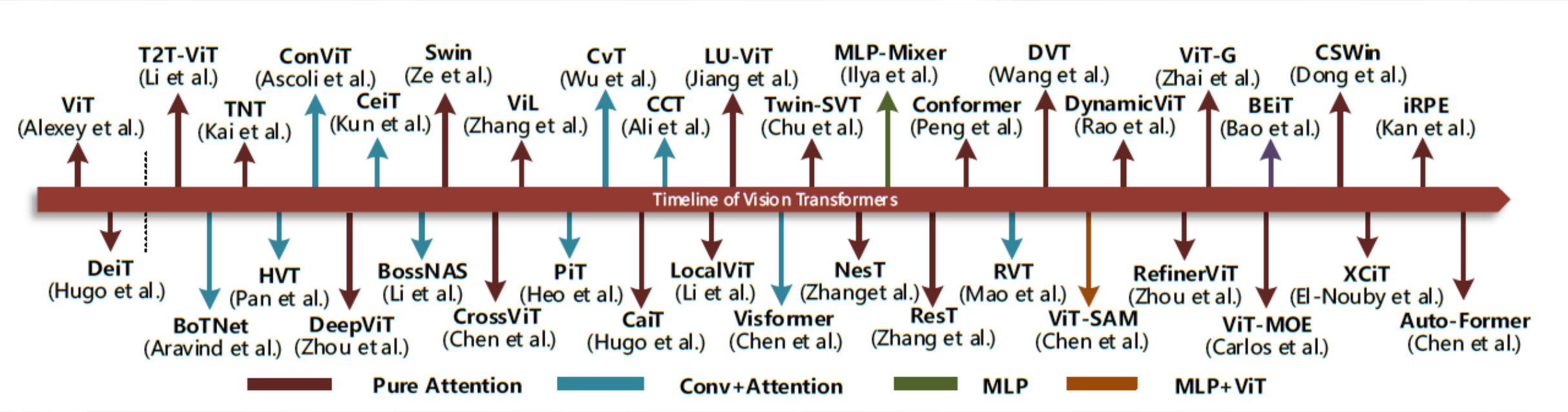
ECANet (CVPR 2020)





Squeeze-and-Excitation Networks. CVPR, 2018.

CNN -> Transformers

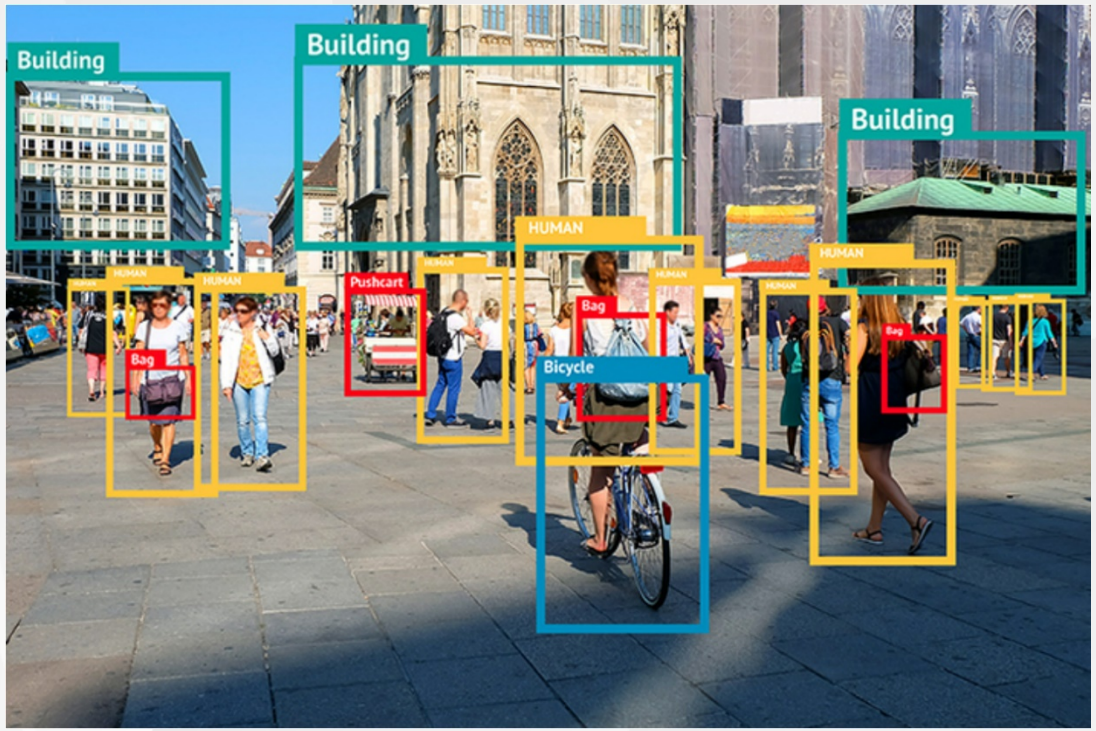


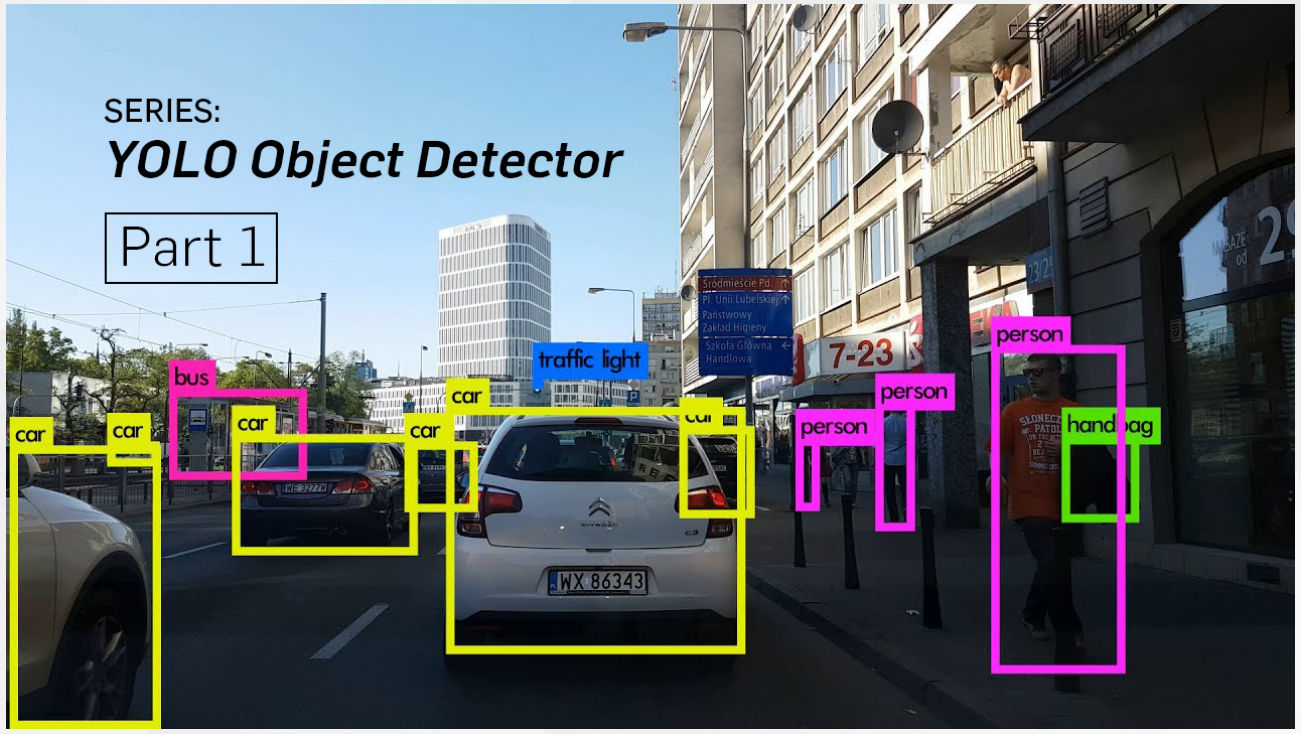
- Swin Transformer

Content

- Introduction
- Low Level Vision: Image Restoration and Generation
- High Level Vision: Visual Understanding
  - Image Classification
  - Object Detection
  - Semantic/Instance Segmentation
- Vision and Language

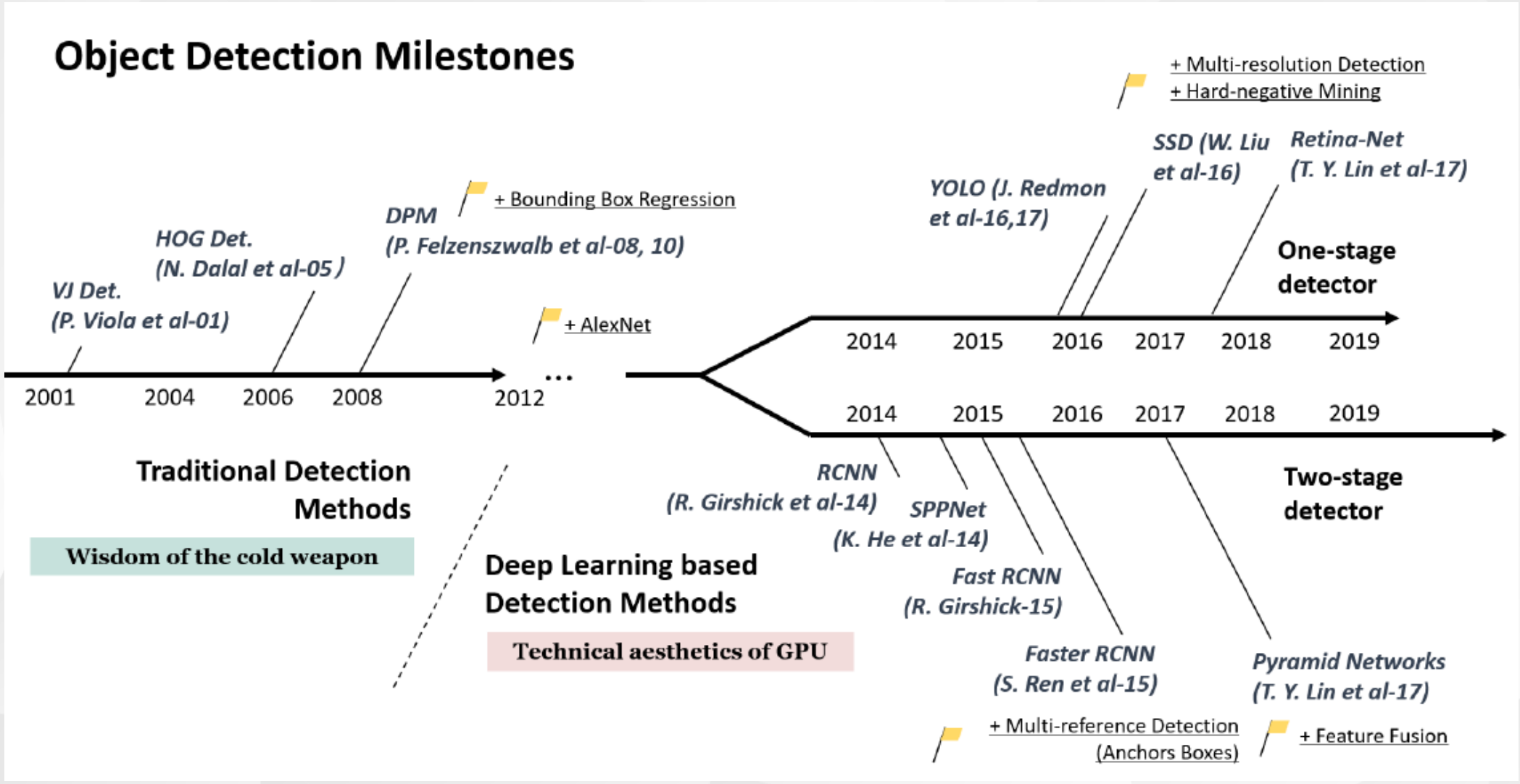
Object Detection



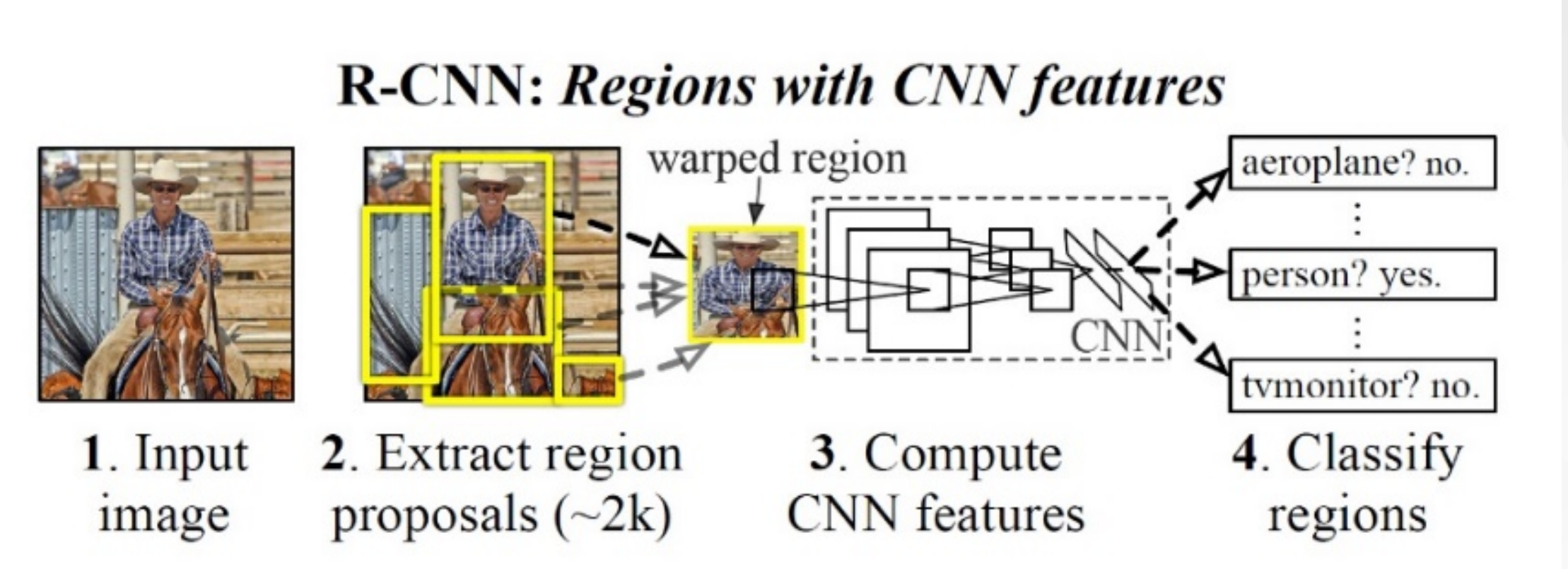


- Classification
- Regression

Object Detection

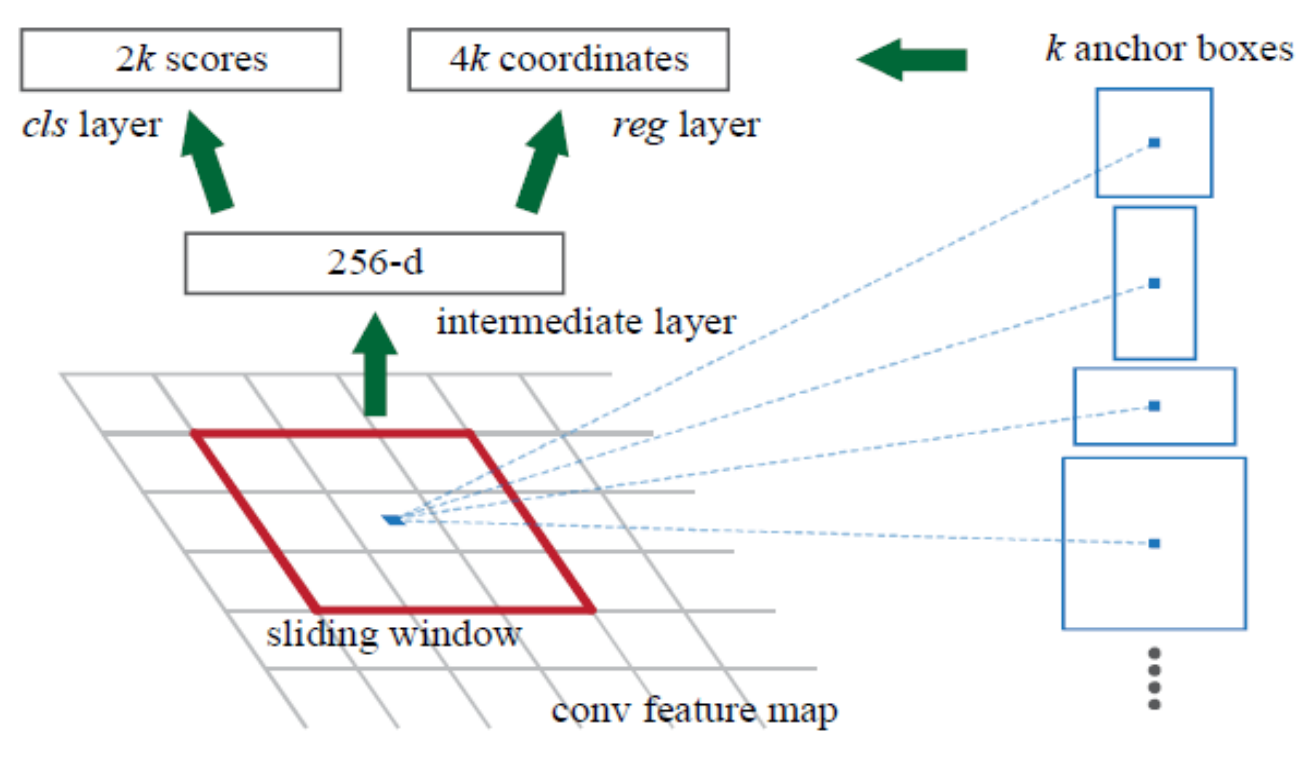


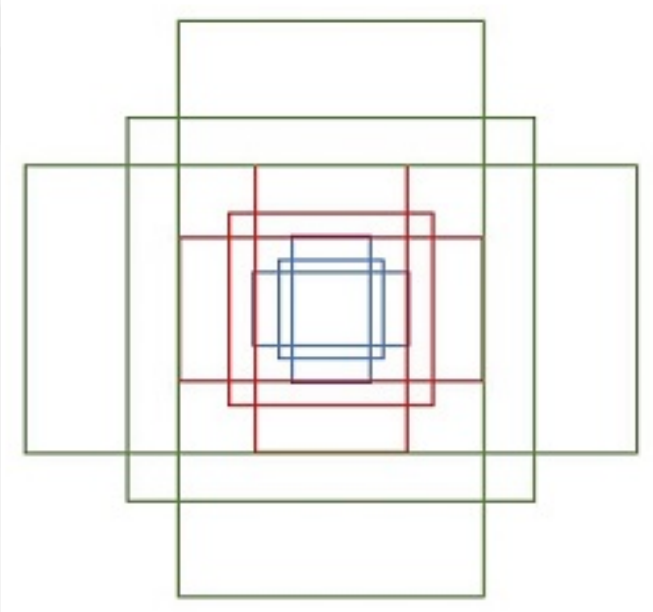
Object Detection: R-CNN



Rich feature hierarchies for accurate object detection and semantic segmentation, CVPR 2014.

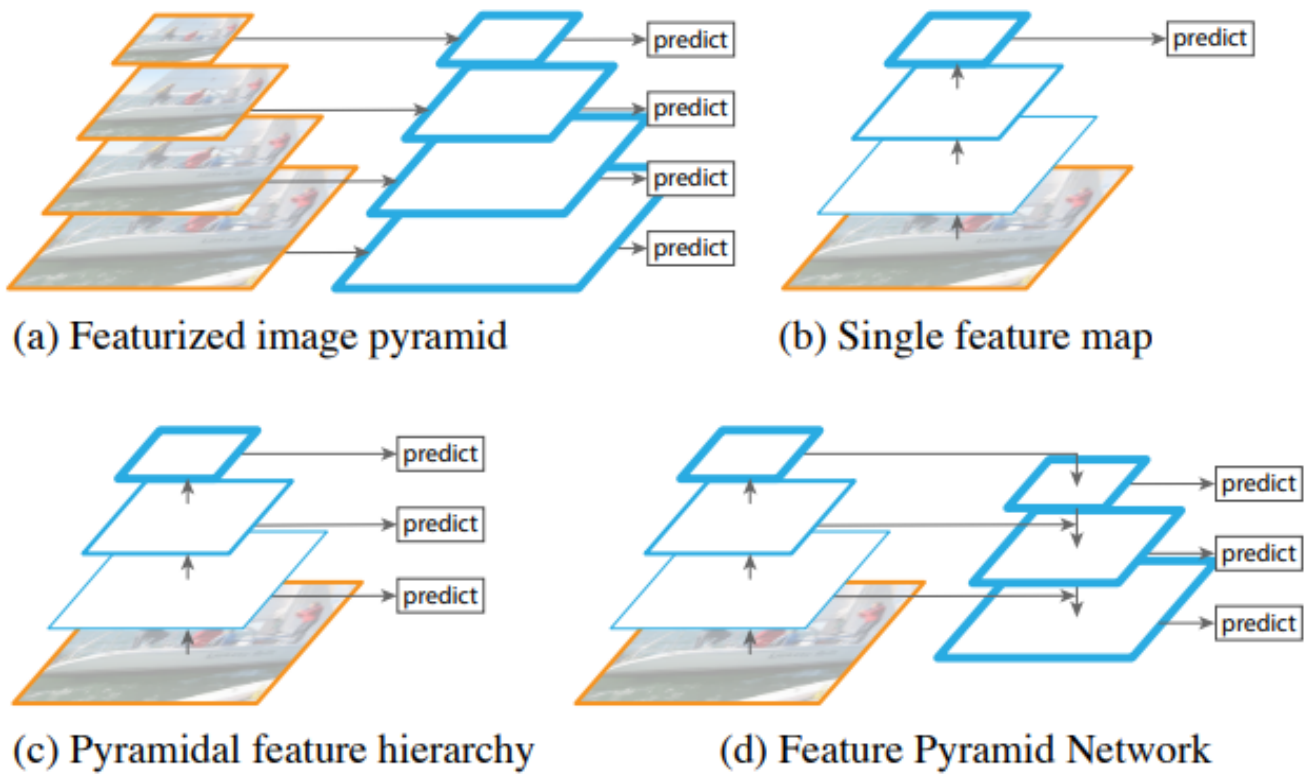
Object Detection: Faster R-CNN

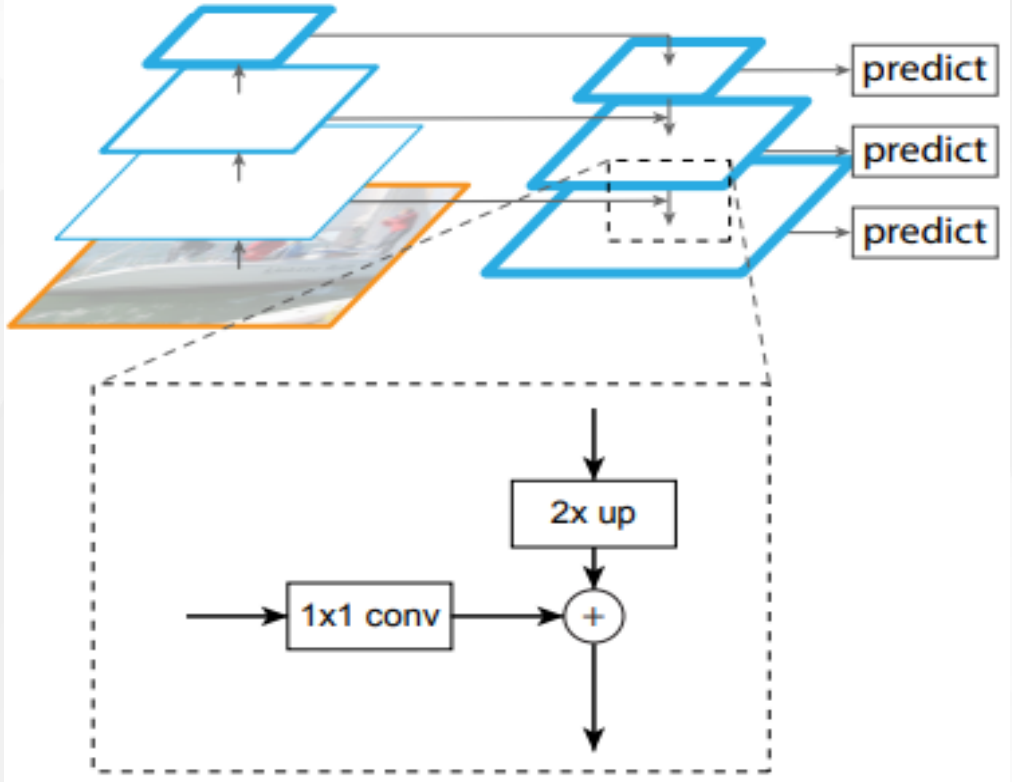




Faster R-CNN: Towards real-time object detection with region proposal networks, NIPS 2015.

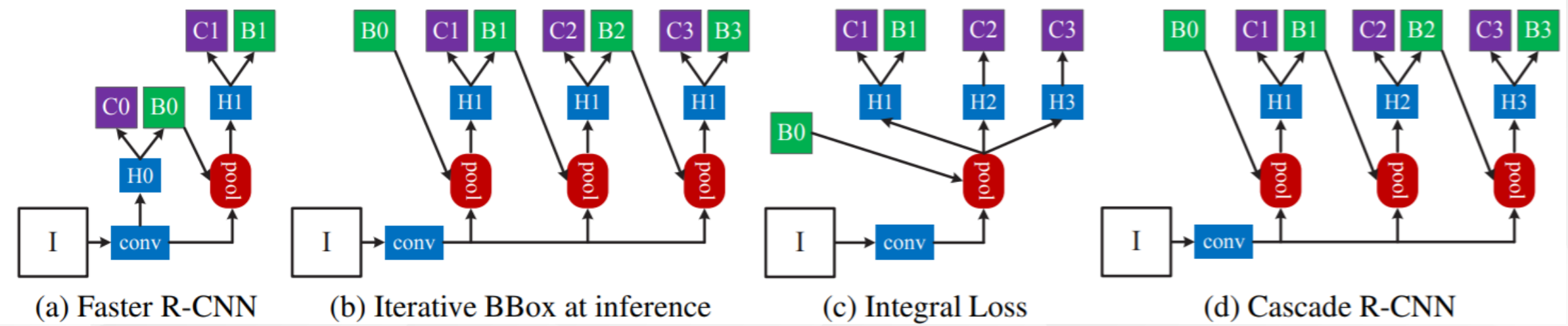
Object Detection: FPN





Feature pyramid networks for object detection, CVPR 2017.

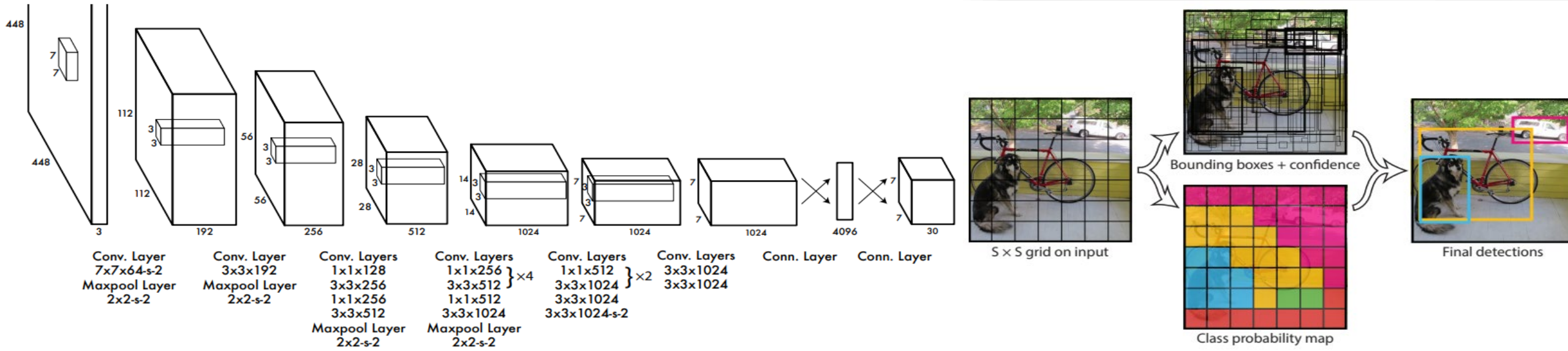
Object Detection: Cascade R-CNN



(a) Faster R-CNN (b) Iterative BBox at inference (c) Integral Loss (d) Cascade R-CNN

Cascade r-cnn: Delving into high quality object detection, CVPR 2018

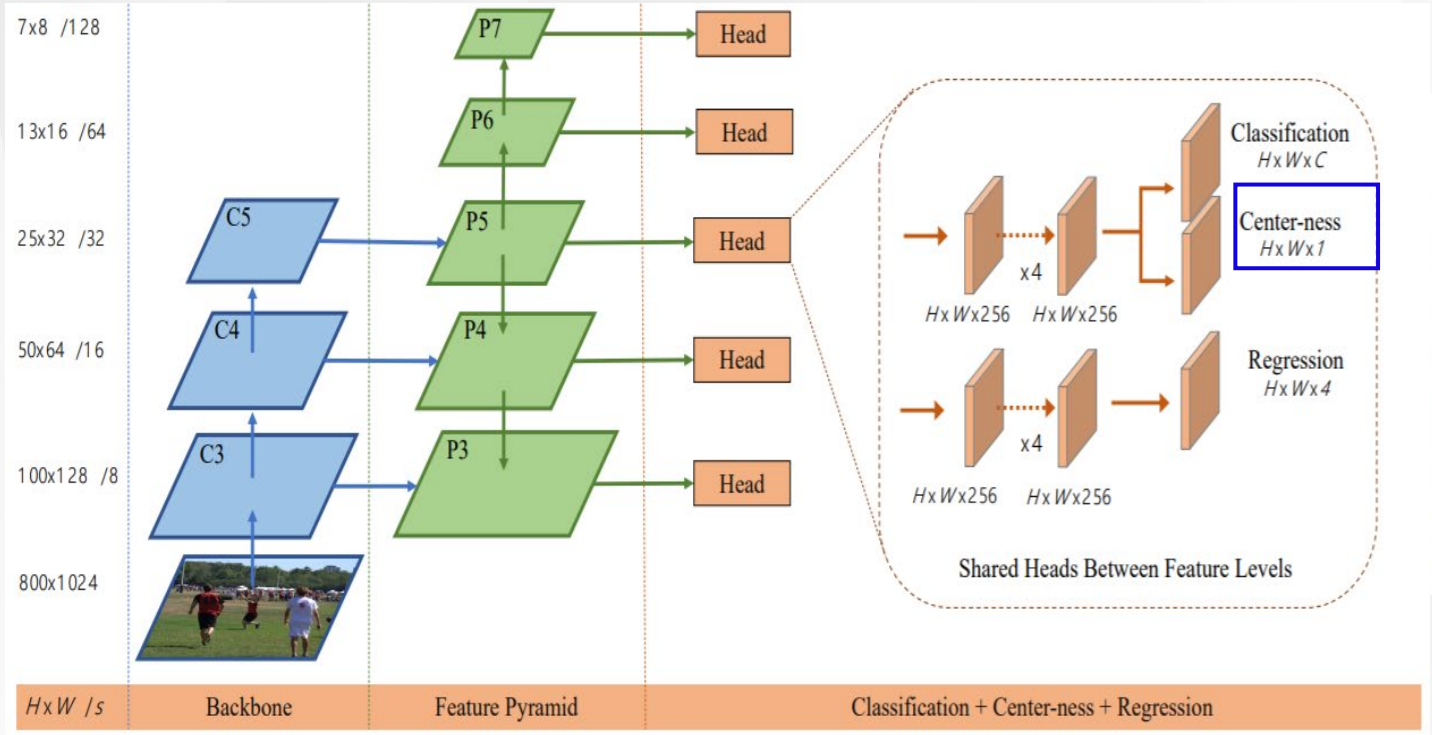
Object Detection: YOLO



$$\Pr(\text{Class}_i \,|\text{Object}) * \Pr(\text{Object}) * \text{IOU}^{\text{truth}}_{\text{pred}} = \Pr(\text{Class}_i) * \text{IOU}^{\text{truth}}_{\text{pred}}$$

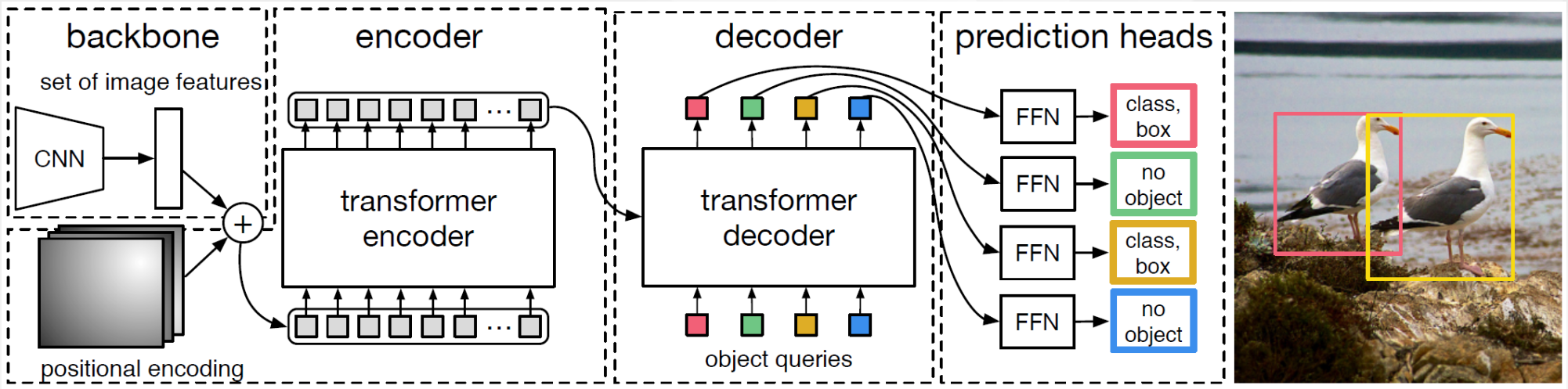
You only look once: Unified, real-time object detection, CVPR 2016.

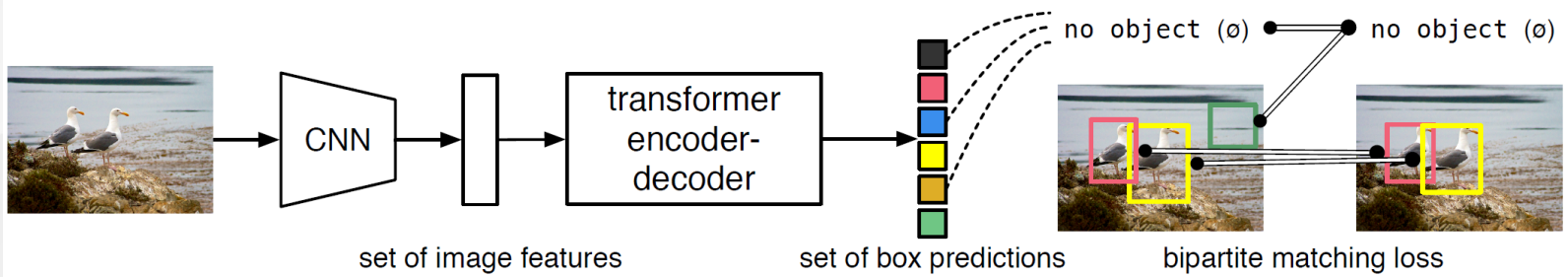
Object Detection: FCOS



FCOS: Fully Convolutional One-Stage Object Detection, ICCV 2019

Object Detection: DETR



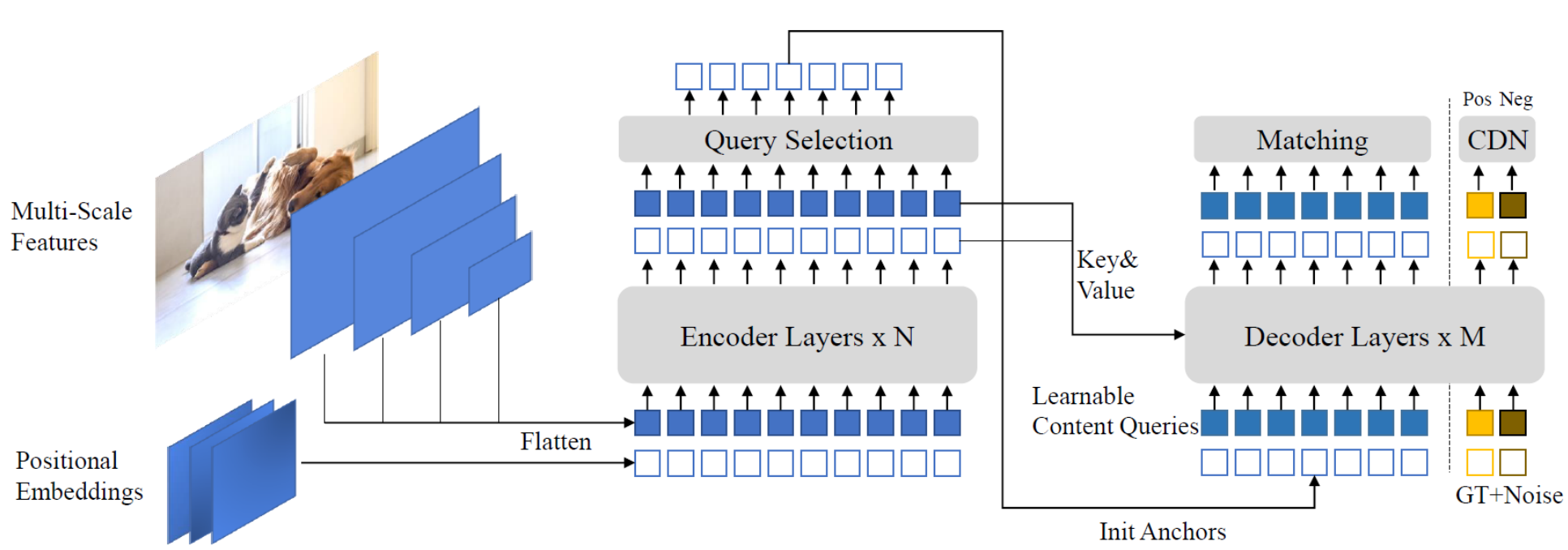


$$\mathcal{L}_{\text{Hungarian}}(y, \hat{y}) = \sum_{i=1}^{N} \left[-\log \hat{p}_{\hat{\sigma}(i)}(c_i) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(b_i, \hat{b}_{\hat{\sigma}}(i)) \right]$$

End-to-end object detection with transformers, ECCV 2020

DINO

- DETR with Improved deNoising anchOr boxes



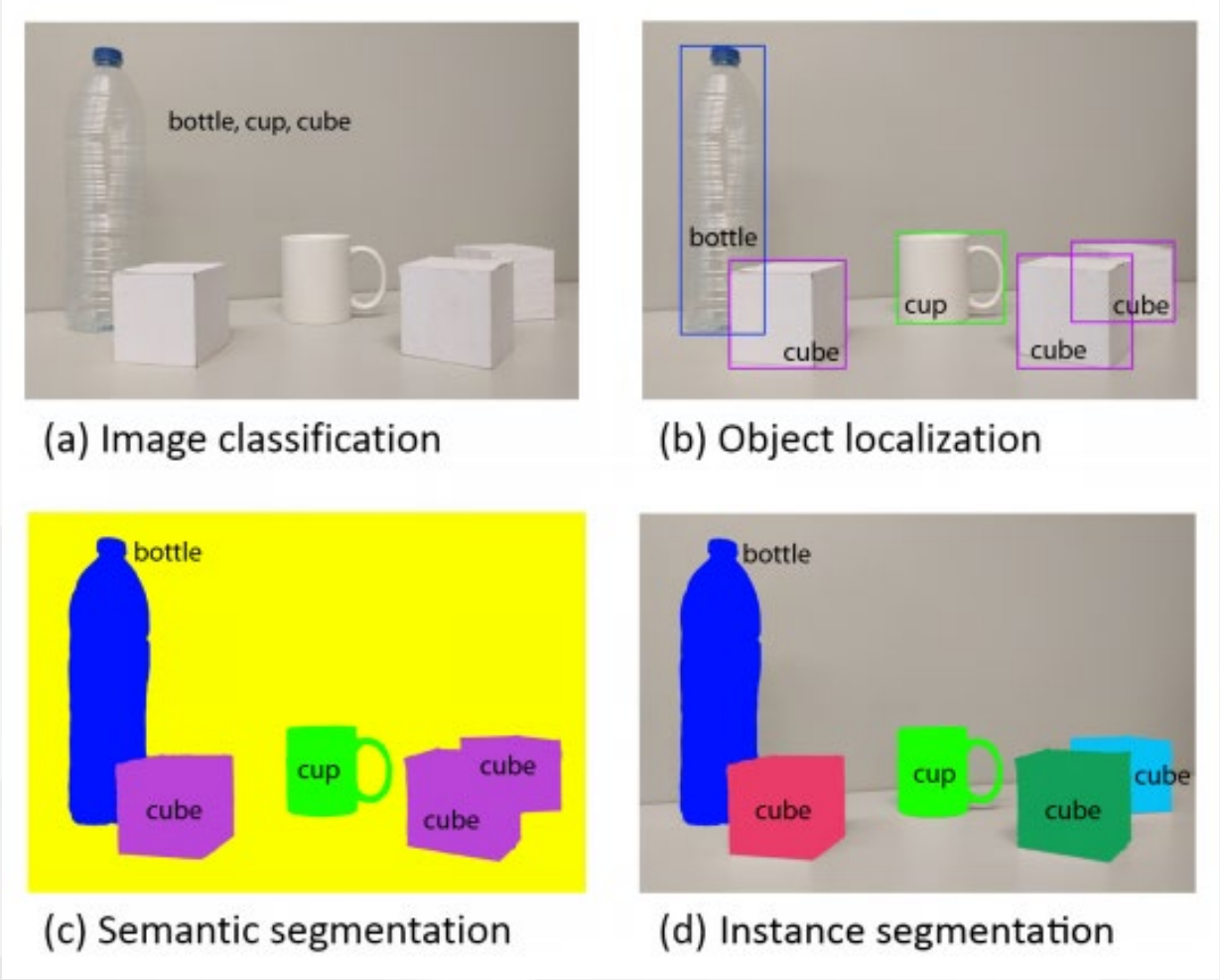
- Contrastive denoising training, look forward twice, and mixed query selection

DINO: DETR with improved denoising anchor boxes for end-to-end object detection, ICLR 2023

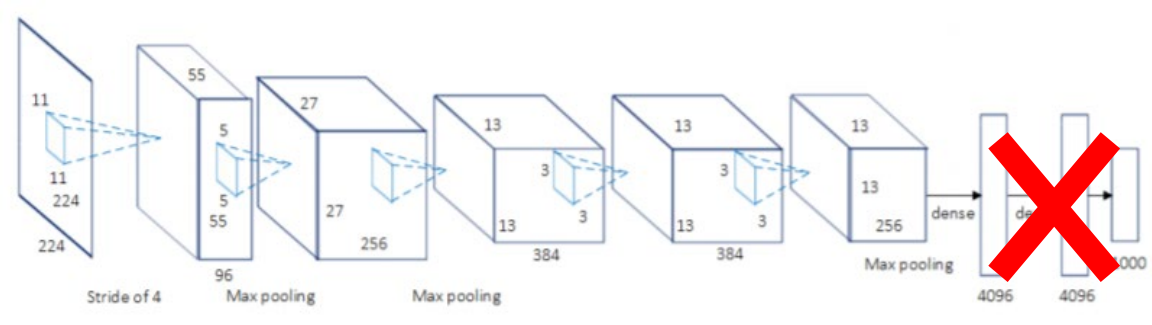
Content

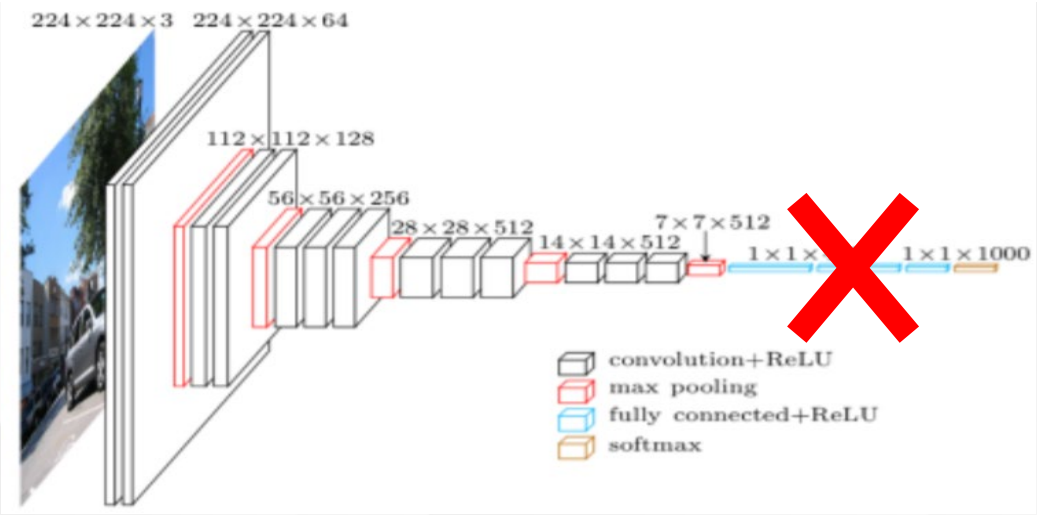
- Introduction
- Low Level Vision: Image Restoration and Generation
- High Level Vision: Visual Understanding
  - Image Classification
  - Object Detection
  - Semantic/Instance Segmentation
- Vision and Language

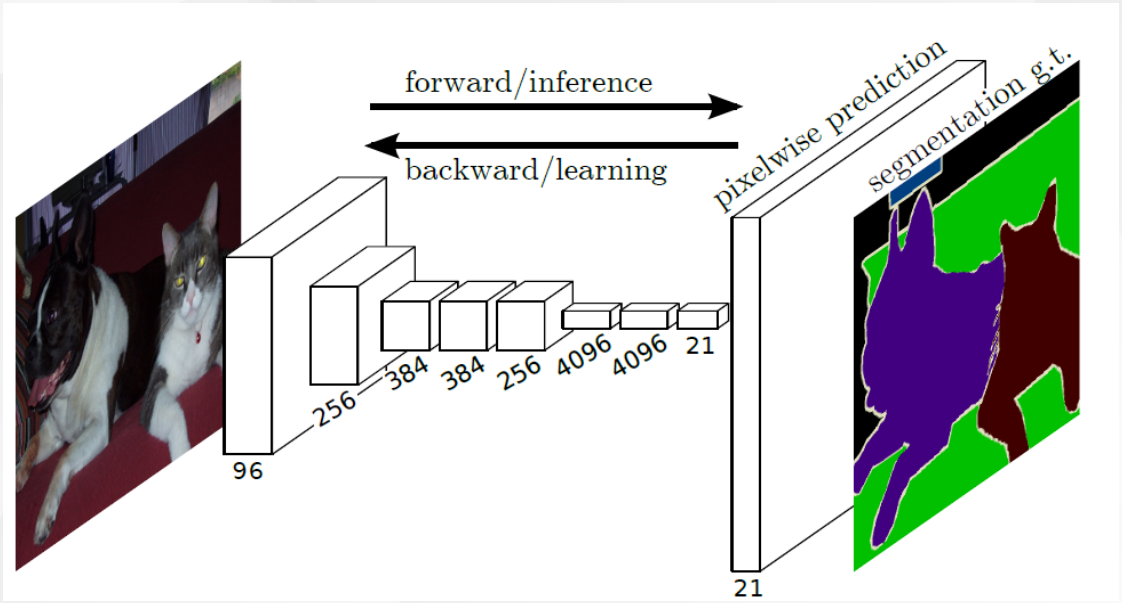
Image Segmentation



Semantic Segmentation: FCN

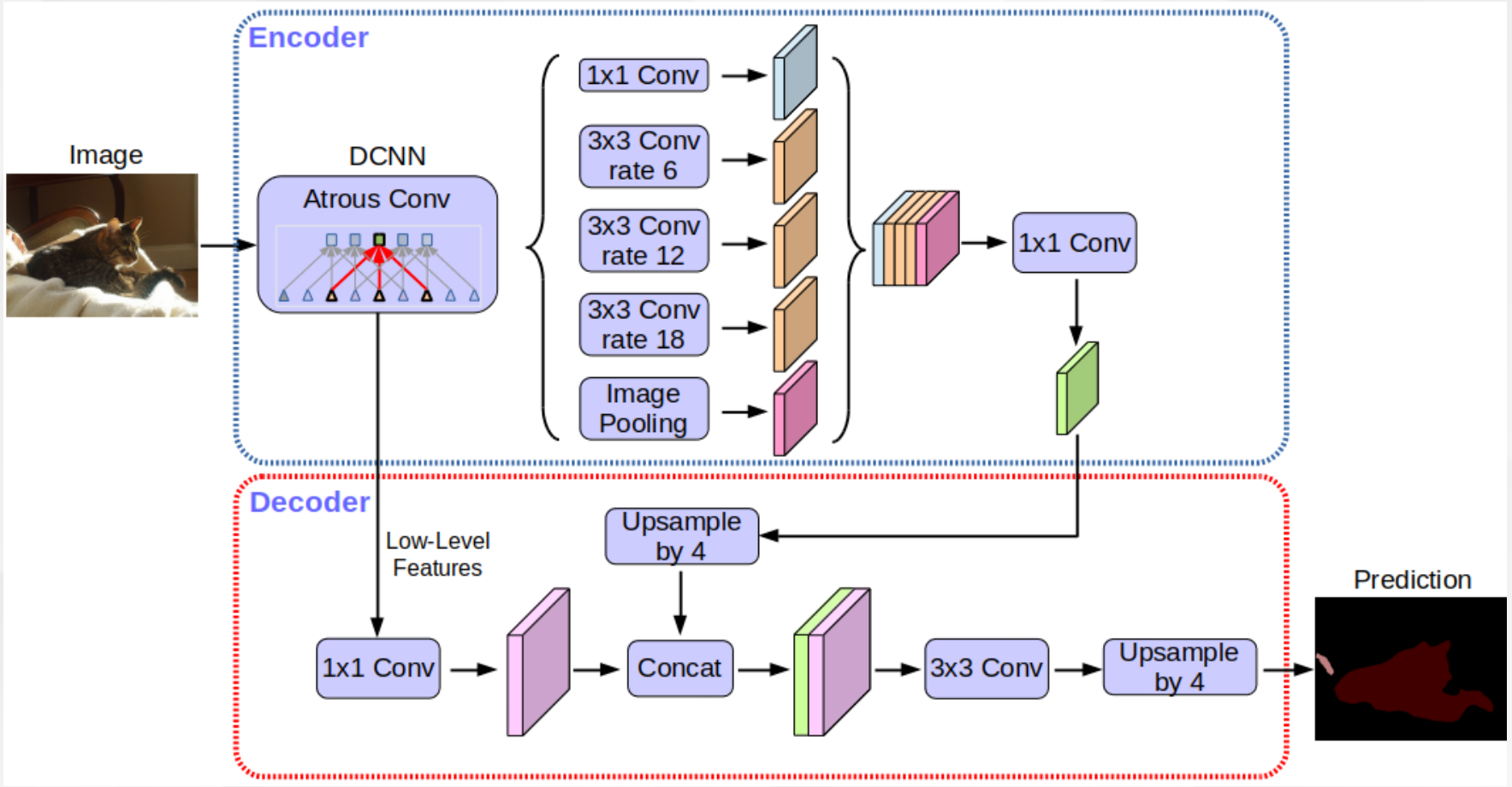






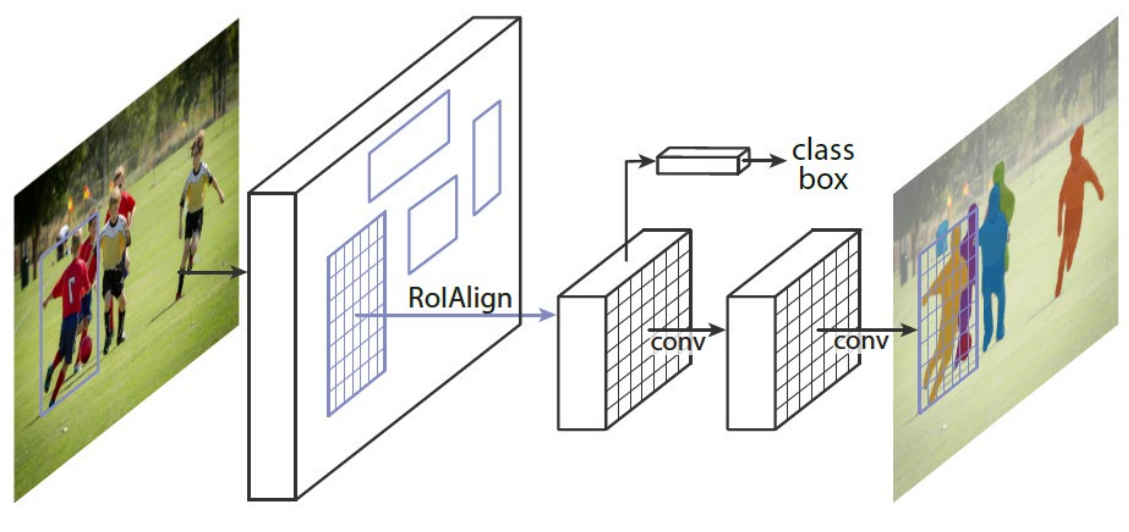
Fully Convolutional Networks for Semantic Segmentation, CVPR 2015

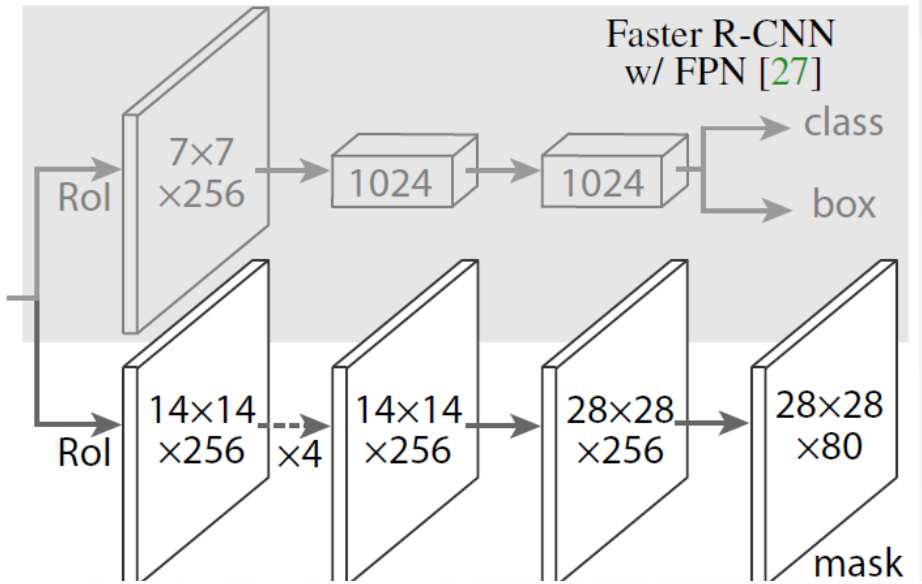
Semantic Segmentation: Deeplab v3

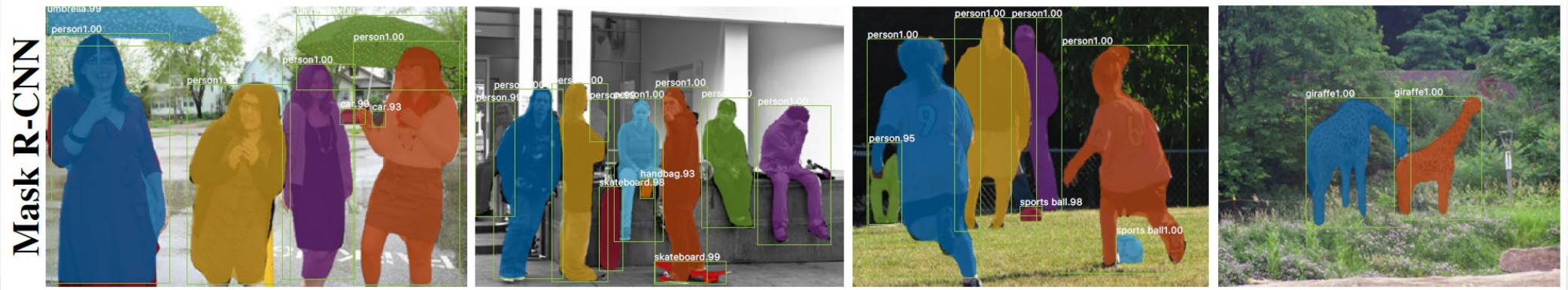


Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation, ECCV 2018.

Instance Segmentation: Mask R-CNN

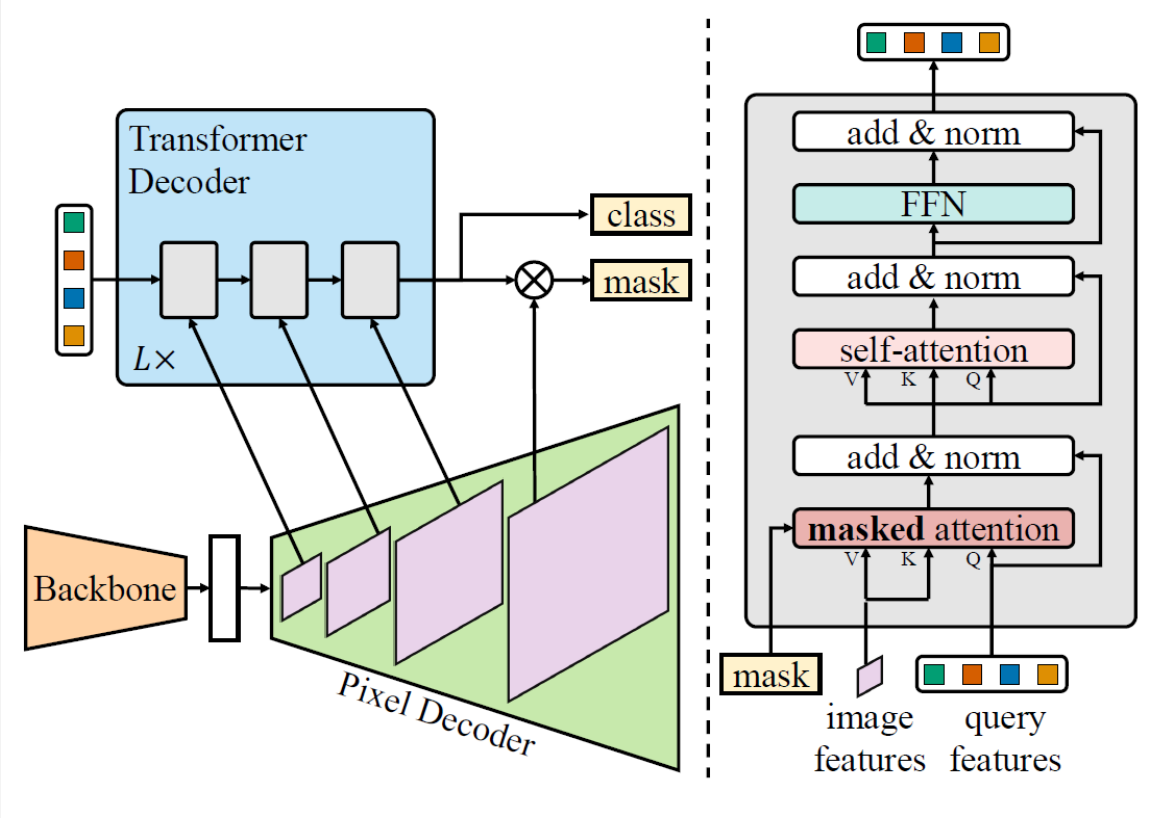


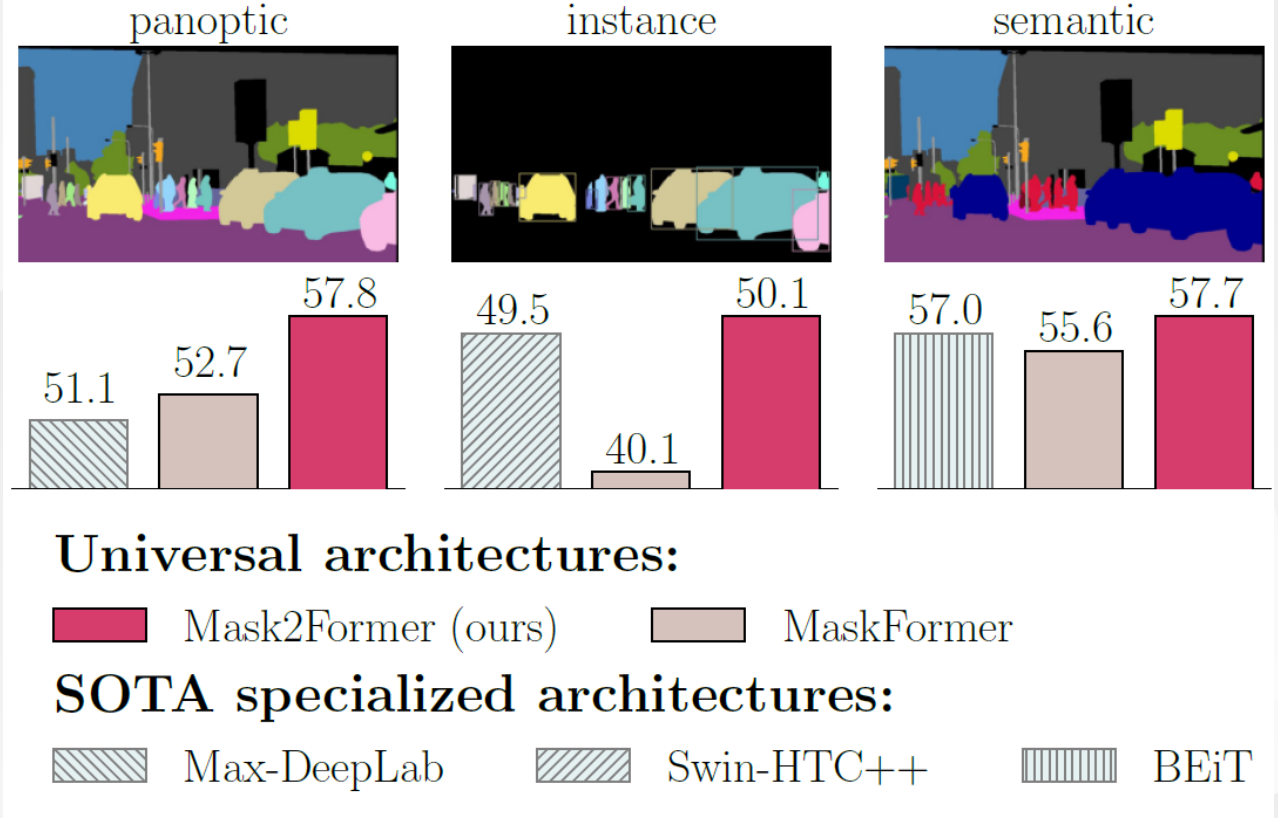




Mask R-CNN, ICCV 2017

Instance Segmentation: Mask2Former





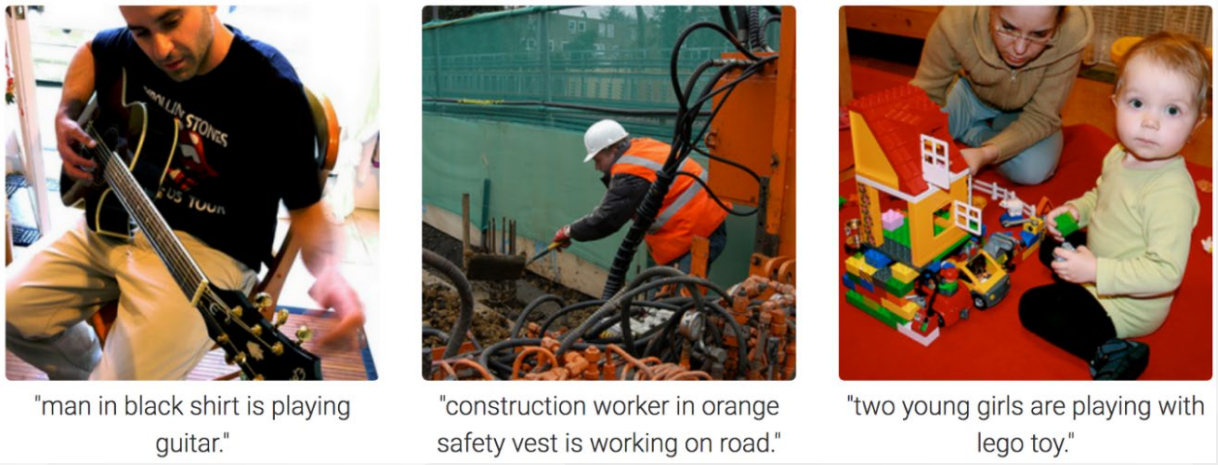
Masked-attention Mask Transformer for Universal Image Segmentation, Arxiv 2021.

Content

- Introduction
- Low Level Vision: Image Restoration and Generation
- High Level Vision: Visual Understanding
- Vision and Language

典型视觉任务（高层视觉）

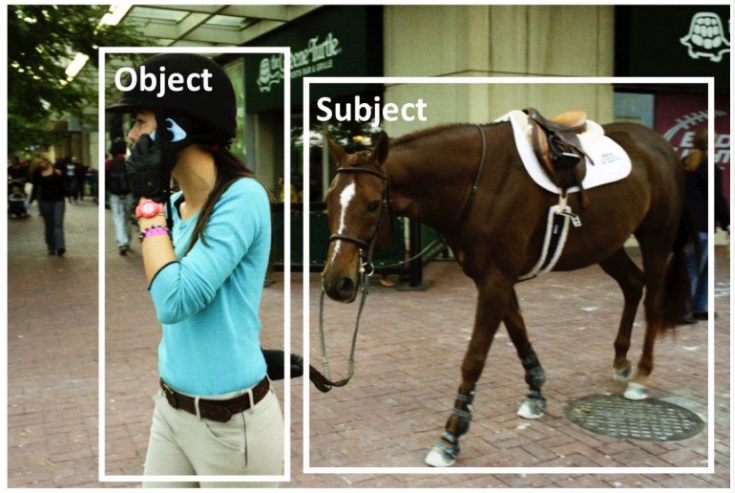
自然语言描述



"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."



指代消解

视觉问答

Q: Does this foundation have any sunscreen?
A: yes

Q: What is this?
A: 10 euros

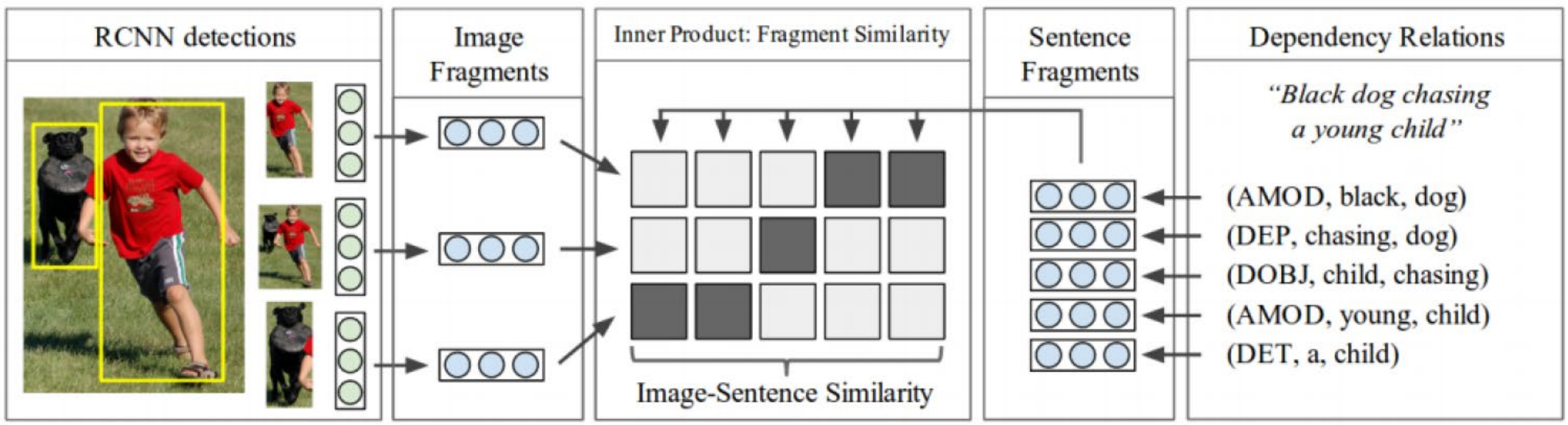
Q: What color is this?
A: green

Q: Please can you tell me what this item is?
A: butternut squash red pepper soup

Q: Is it sunny outside?
A: yes

Q: Is this air conditioner on fan, dehumidifier, or air conditioning?
A: air conditioning

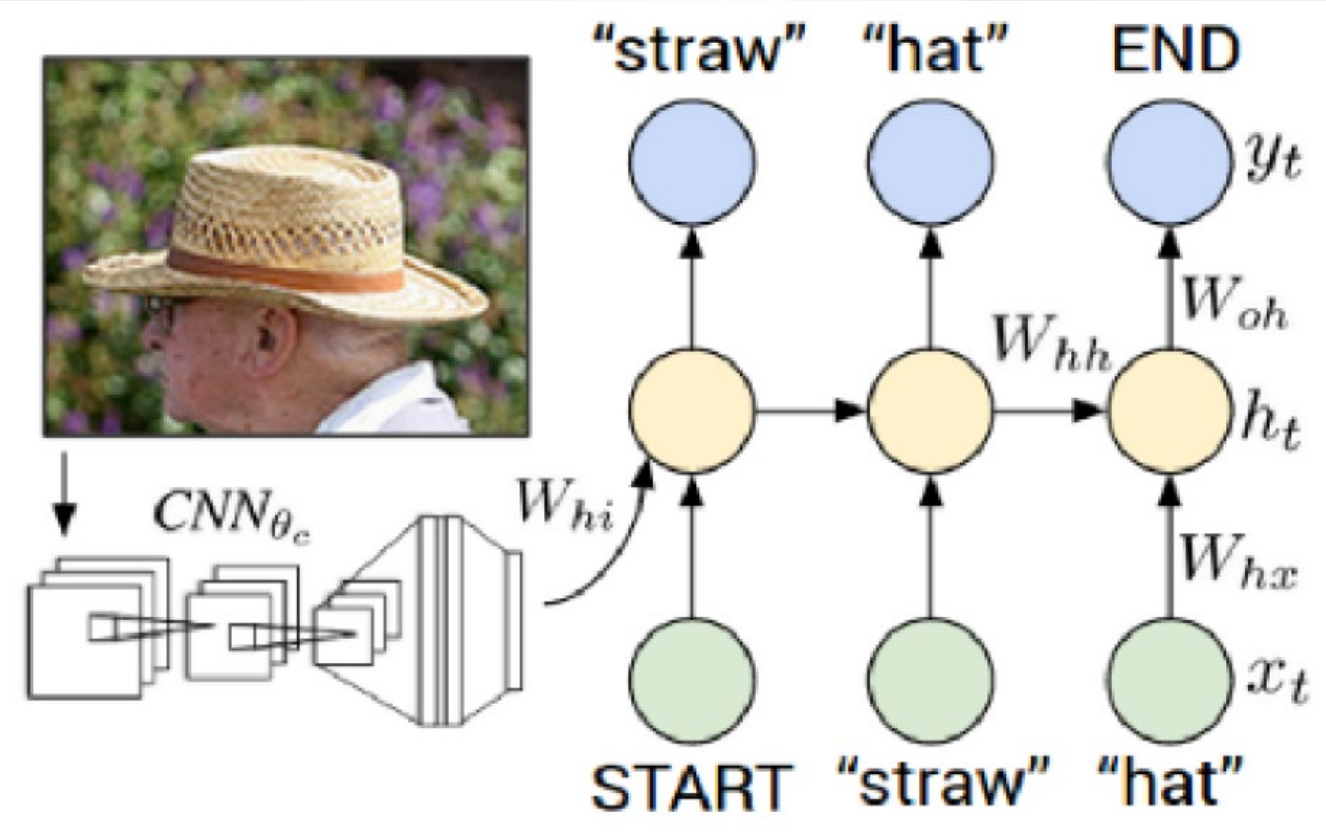
Image Text Matching



Deep fragment embeddings for bidirectional image sentence mapping, NIPS 2014.

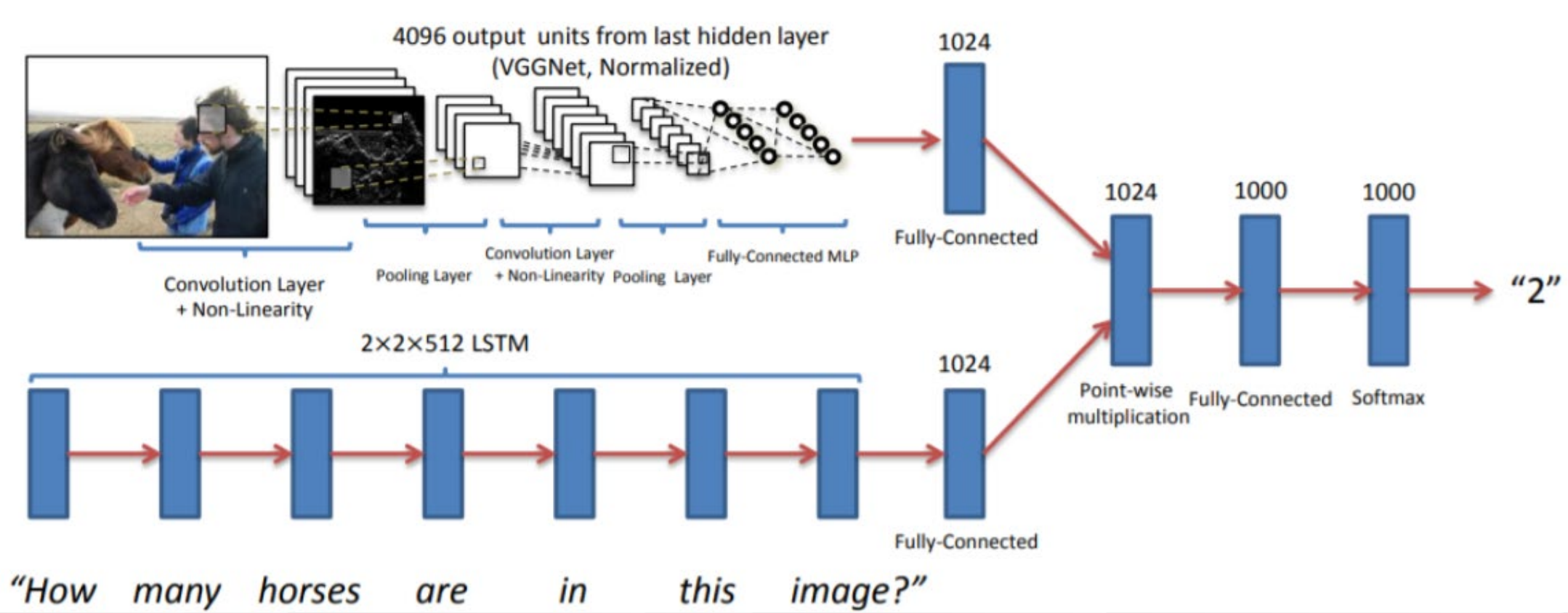
Image Captioning

- CNN+LSTM



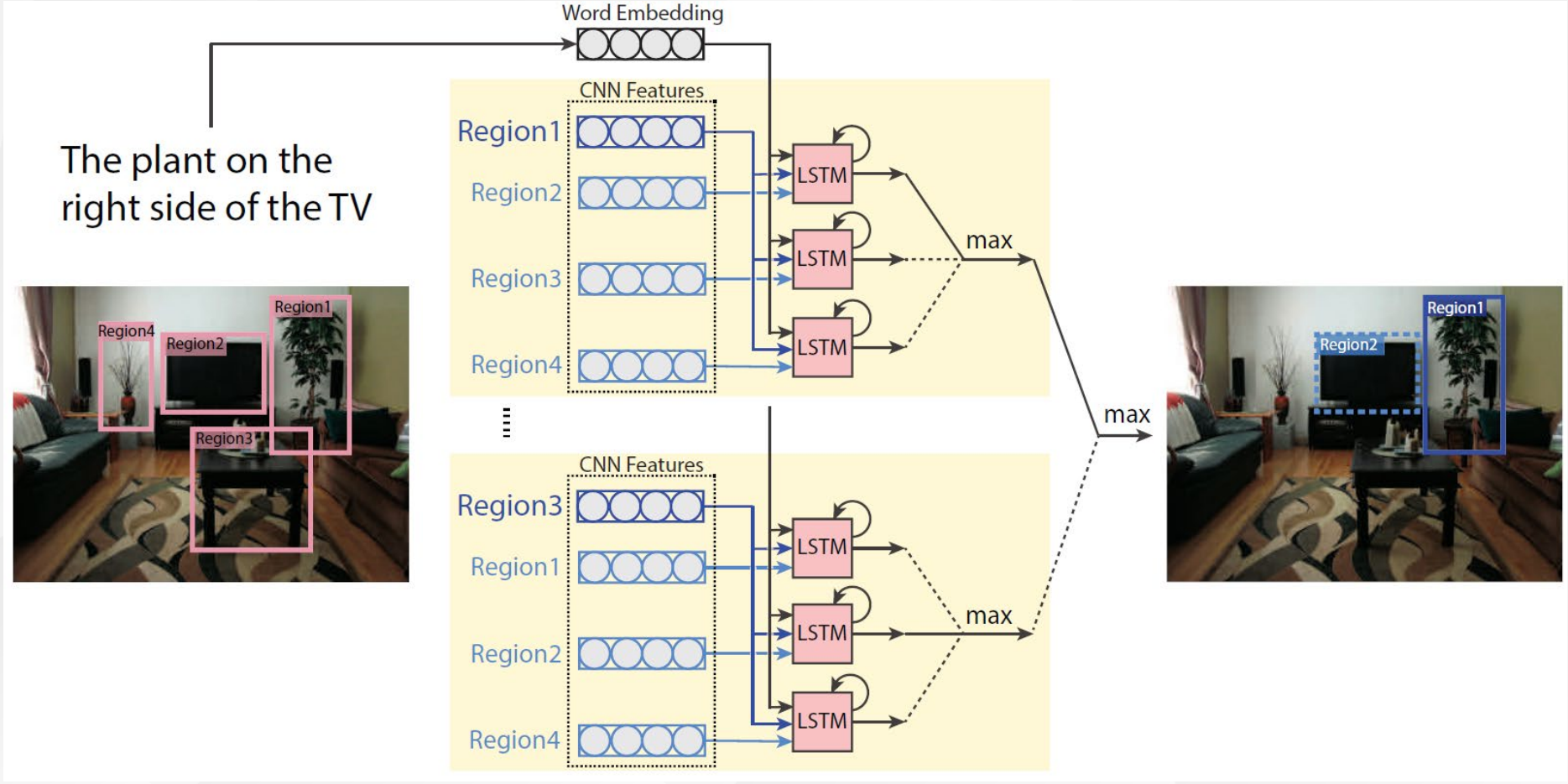
Deep visual-semantic alignments for generating image descriptions, CVPR 2015.

Visual Question Answering



Vqa: Visual question answering, ICCV 2015.

Visual Grounding



Modeling Context Between Objects for Referring Expression Understanding, ECCV 2016

Summary

- 如何结合具体的视觉任务，设计恰当的深度网络和学习目标：
  - 深度网络的进展
  - 对具体视觉任务的认识
  - 对具体数据的认识

  - 其他视觉数据类型：3D、视频
  - 全监督学习 -> 其他监督学习方式、预训练+Finetuning

Course Arrangement

- 7. CNNs -> Transformers (2)
- 8. Vision Tasks and Learning (2)
- 9. Self-Supervised Learning (2)
- 10. V-L Pretraining and Downstream Tasks (2)
- 11. Generative Adversarial Networks (2)
- 12. Text-to-Image Generation and Downstream Tasks (2)

Thank You !