Convolutional Neural Networks

Viorica Pătrăucean

DeepMind

Outline

CNNs Intro: Taxonomy, What, Why

Layers: Conv, Pooling, Unpooling, BatchNorm

Tasks, Models, Datasets

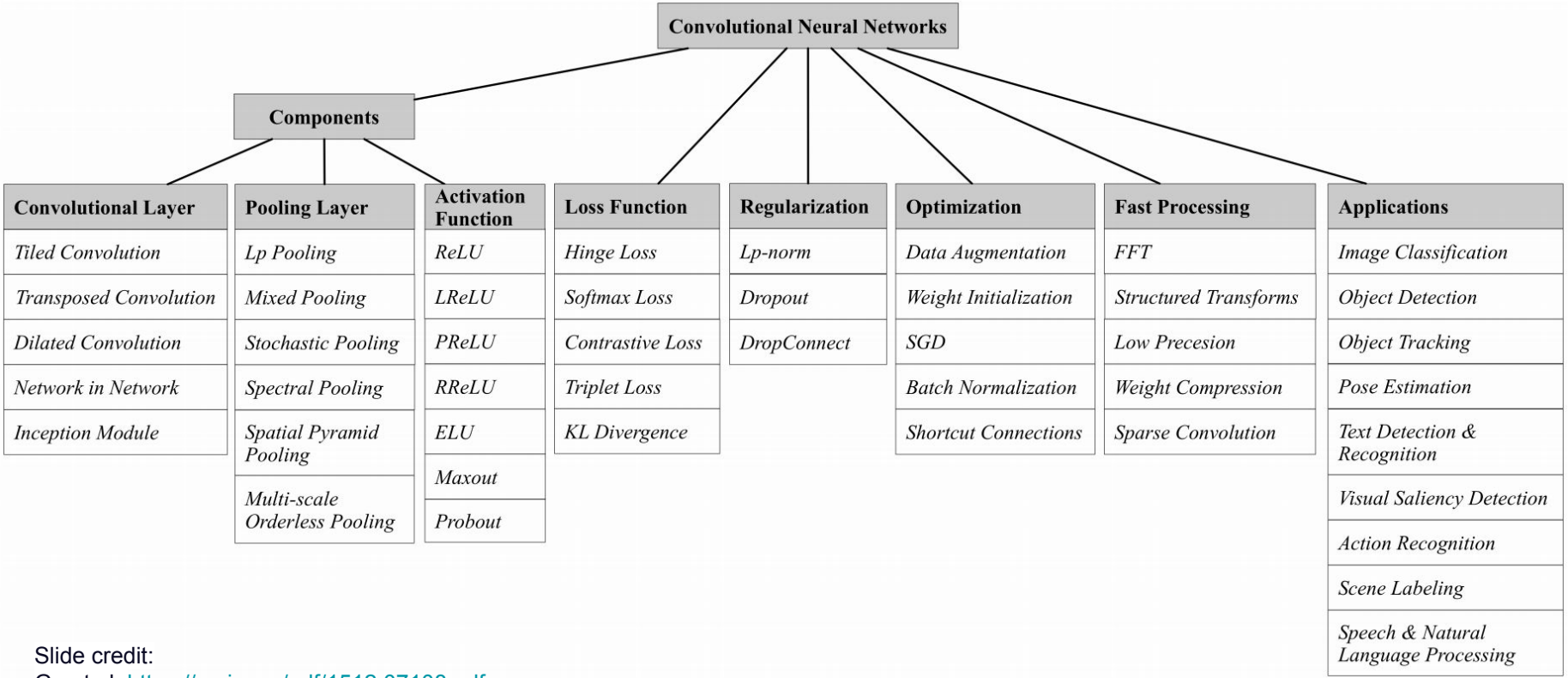
Learning representations

Beyond images: video, sound

Beyond CNNs: Graphs Convnets, Transformers

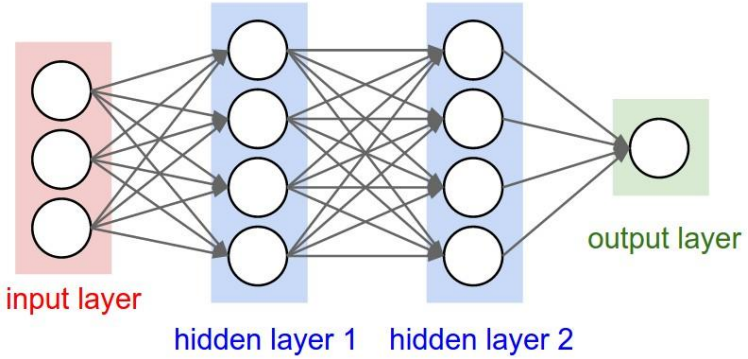
Introduction

Taxonomy



Slide credit:
Gu et al. https://arxiv.org/pdf/1512.07108.pdf
Shrivastava https://umd.app.box.com/s/wbhdfdkbnlw92k08edw77xcdlsxm4mkd

CNNs vs MLPs



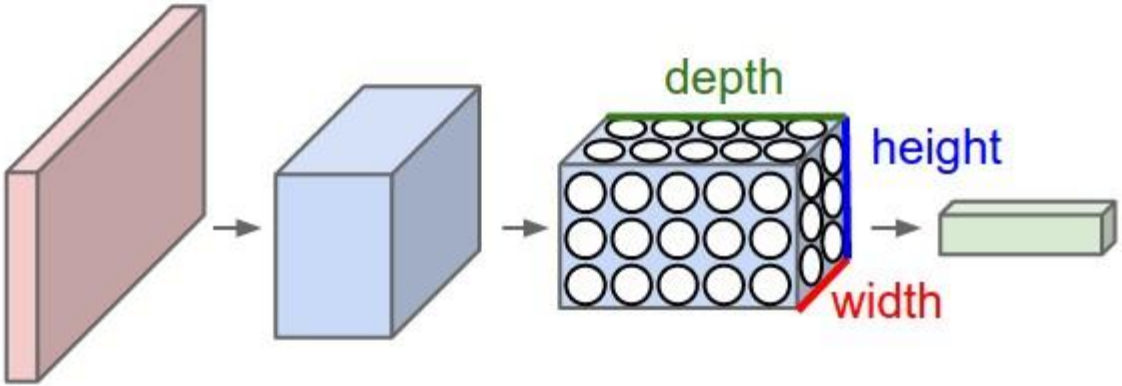


Diagram credit:
Karpathy http://cs231n.github.io/convolutional-networks/

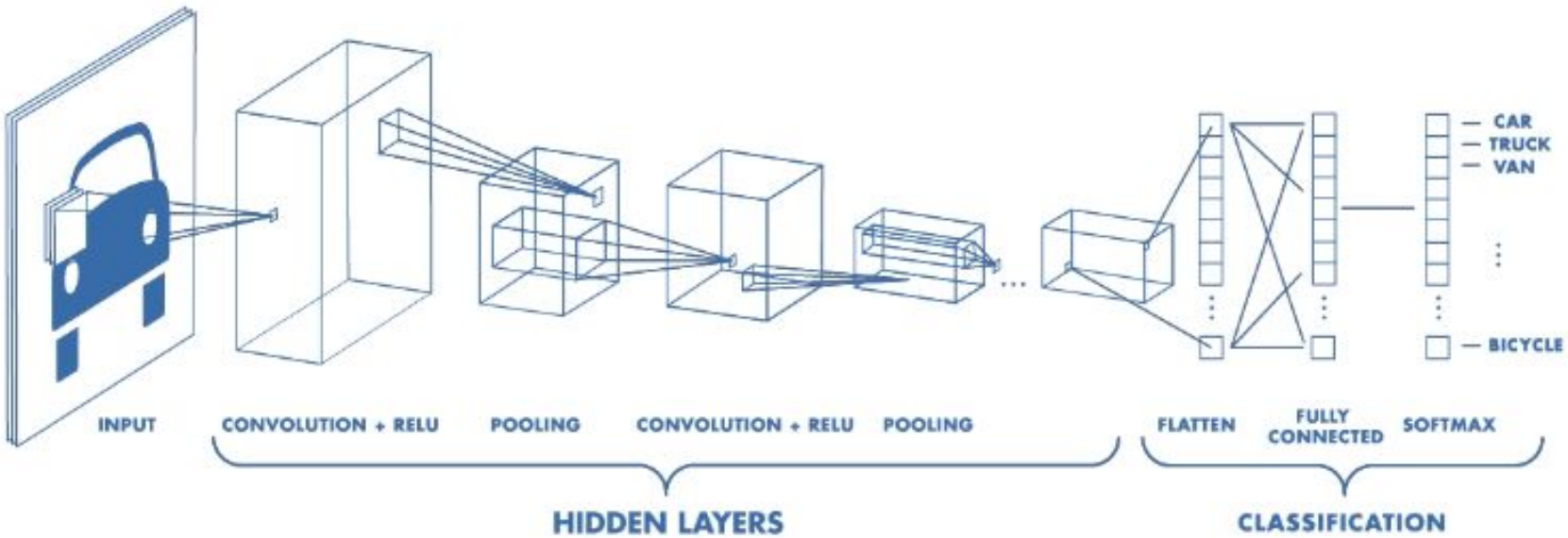
Locality of the data & Spatial invariance

Hierarchical structure

Shared features across classes (edges, local shape cues)

More abstract class-specific features built on top of simple shared features

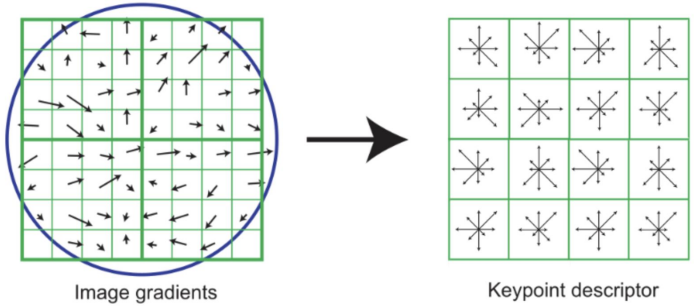
Convnets: adding structure to models



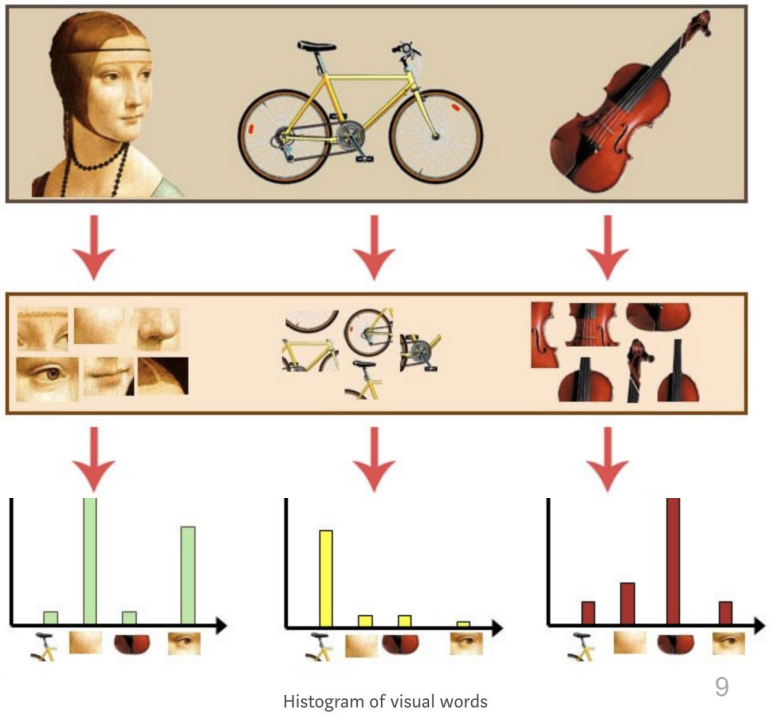
Slide credit:
https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html
https://www.freecodecamp.org/news/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050/

Old school Computer Vision: SIFT & Bag of Words

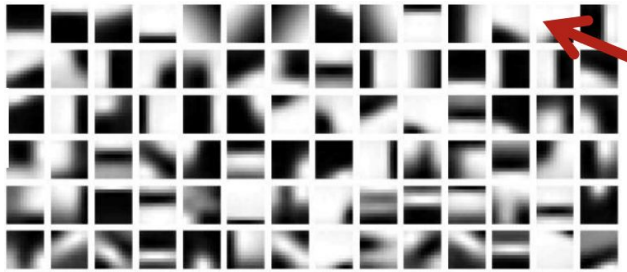
SIFT - D. Lowe, 1999



Bag of visual words - Sivic et al, 2005

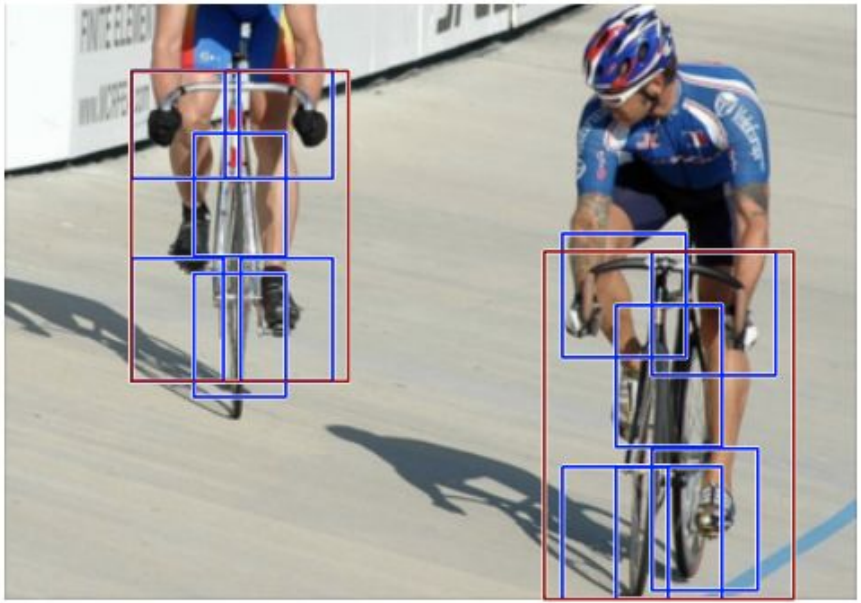


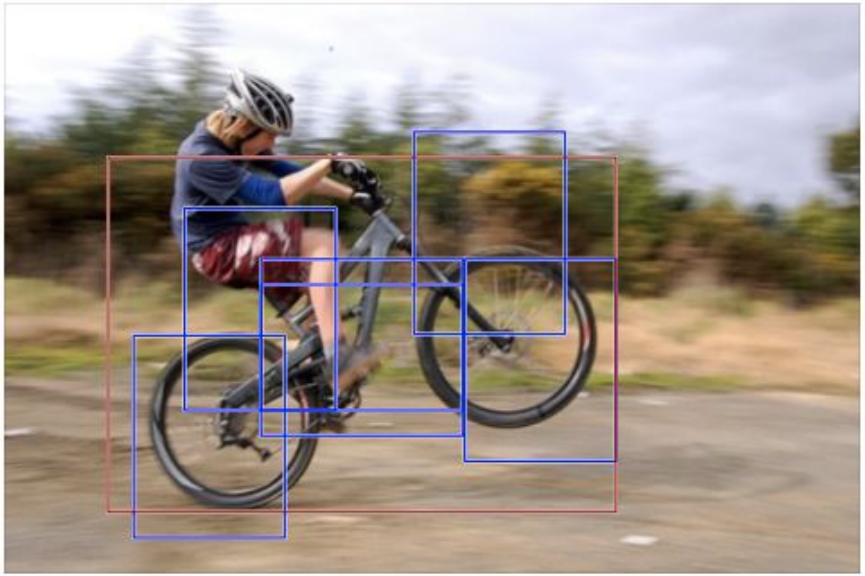
Codebook



Visual word

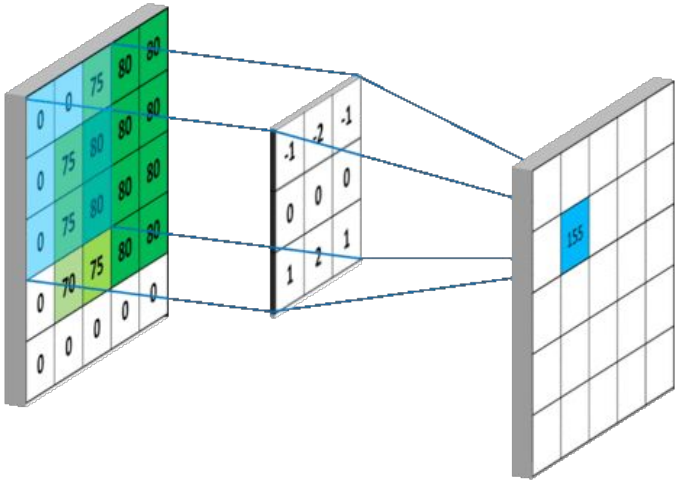
Old school Computer Vision: Part-based models





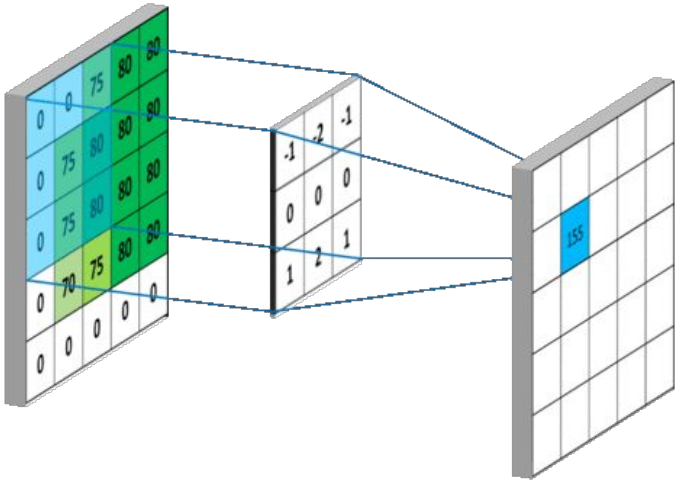
Deformable part-based models, 2010

Data locality -> local connectivity



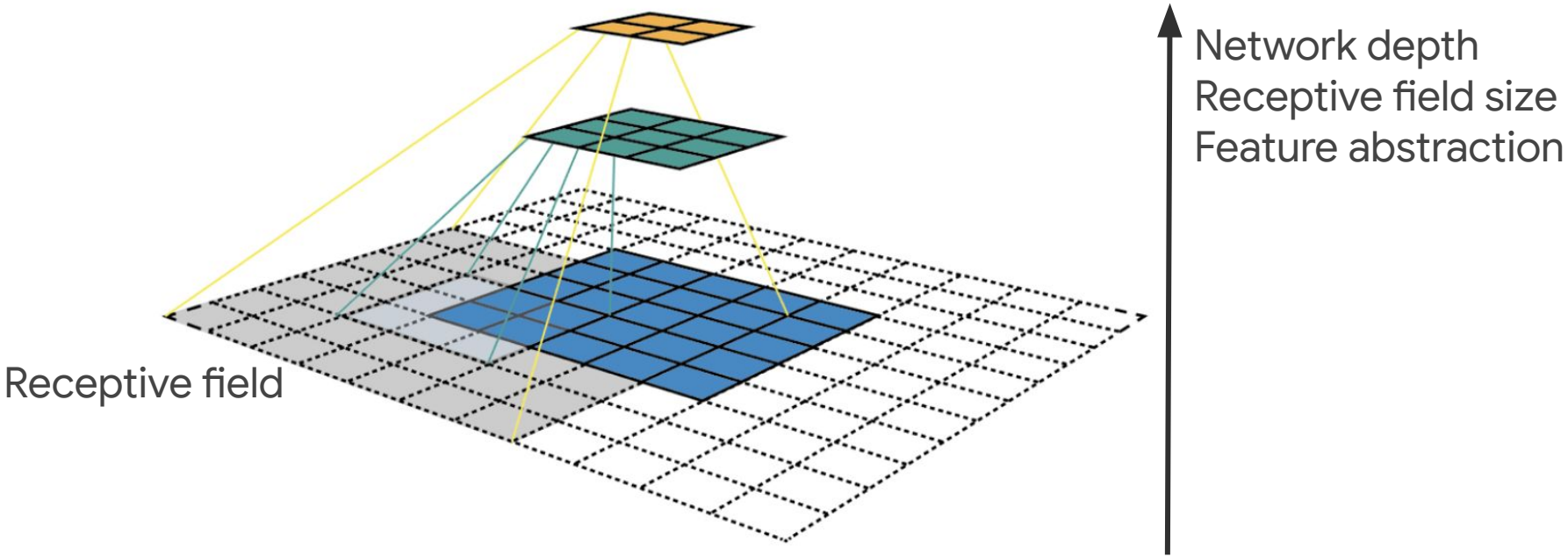
Spatial neighbourhood defines the role of a pixel

Data locality -> local connectivity



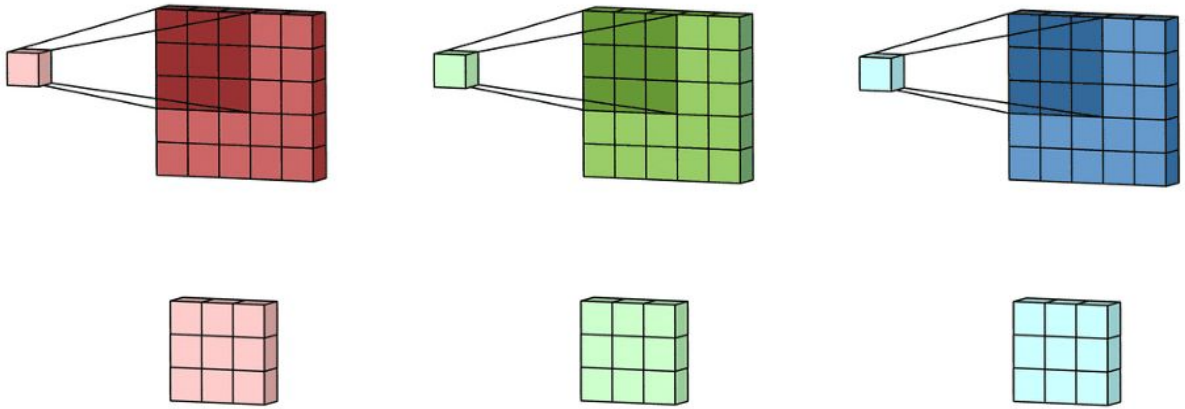
Translation invariance -> weights sharing

Hierarchical structure: the deeper the better



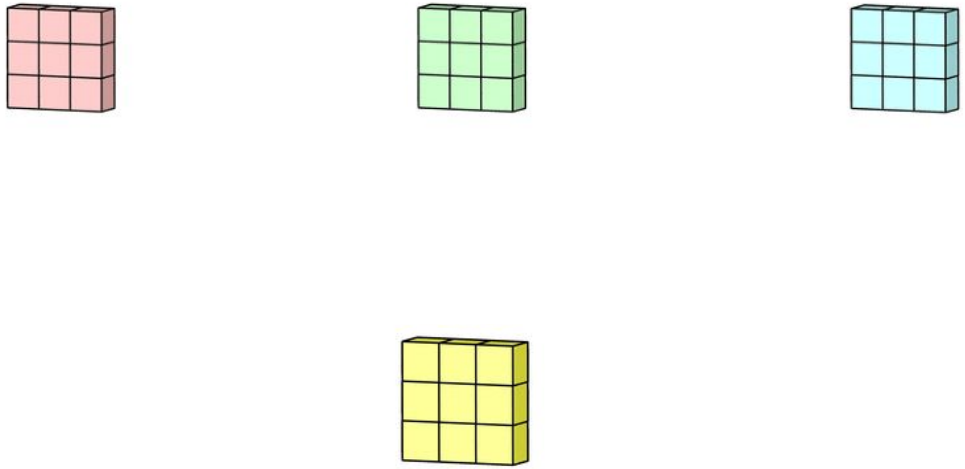
Slide credit:
https://medium.com/mlreview/a-guide-to-receptive-field-arithmetic-for-convolutional-neural-networks-e0f514068807

Multiple feature maps: kernel vs filter



Slide credit:
Bai https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215

Multiple feature maps: kernel bias



Slide credit:
Bai https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215

Summary

Inductive biases (Relational inductive biases, deep learning, and graph networks, Bataglia et al)

- Structure introduced in the model based on **assumptions about data** and **task requirements** (domain knowledge);
- Restricts the search space for models; makes optimization easier
- Convnets inductive biases:
  - **Hierarchical representation**: abstraction increases with depth and size of receptive field
  - **Locality of the data** -> local filters
  - **Translation invariance** -> weights sharing.
  - Reduced number of parameters.

Layers

Convolutional layer forward pass

Input feature map: 3x3, filter 2x2, stride: 1, padding: VALID => output 2x2

Im2col - efficient matrix to matrix multiplication

https://becominghuman.ai/back-propagation-in-convolutional-neural-networks-intuition-and-code-714ef1c38199

Convolutional layer backward pass

∂h_{ij} represents $\frac{\partial L}{\partial h_{ij}}$

∂w_{ij} represents $\frac{\partial L}{\partial w_{ij}}$

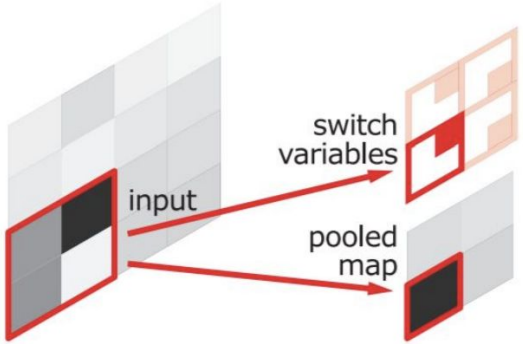
Notations

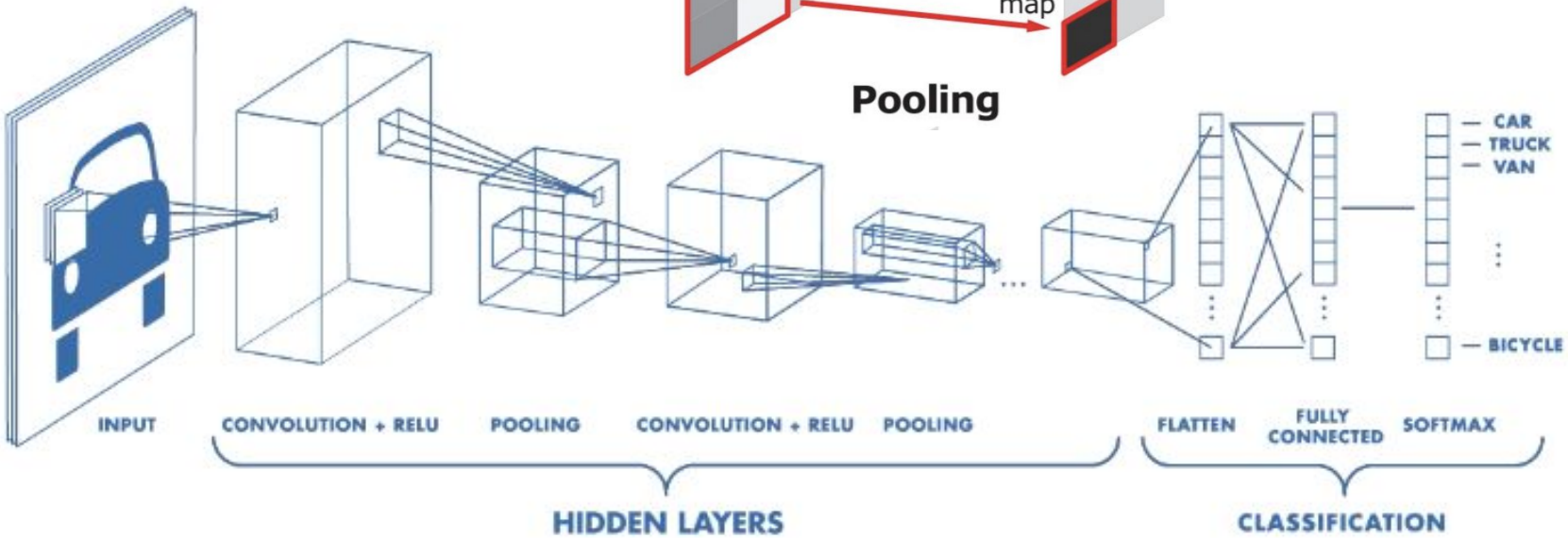
Derivatives wrt conv layer weights: sum contributions from all incoming derivatives, due to shared weights.

Derivatives wrt input: sum over corresponding window.

https://becominghuman.ai/back-propagation-in-convolutional-neural-networks-intuition-and-code-714ef1c38199

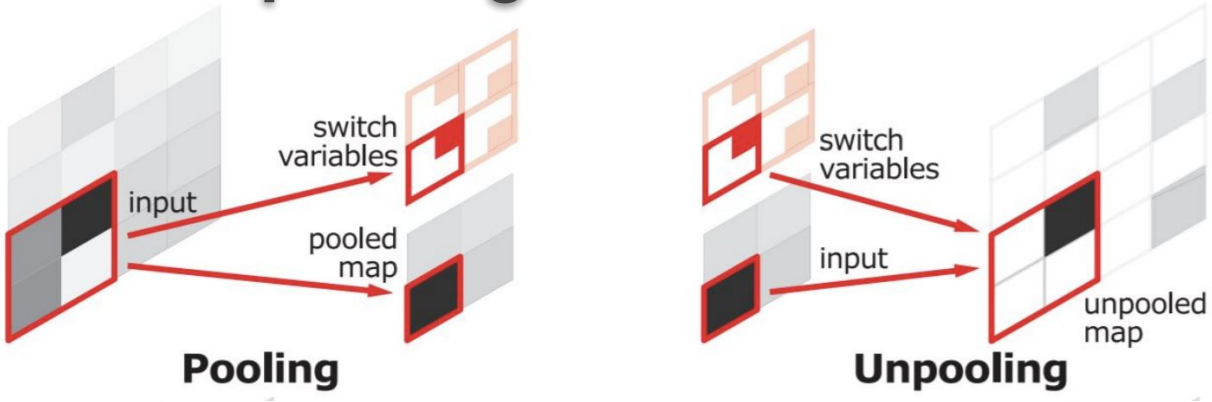
Pooling

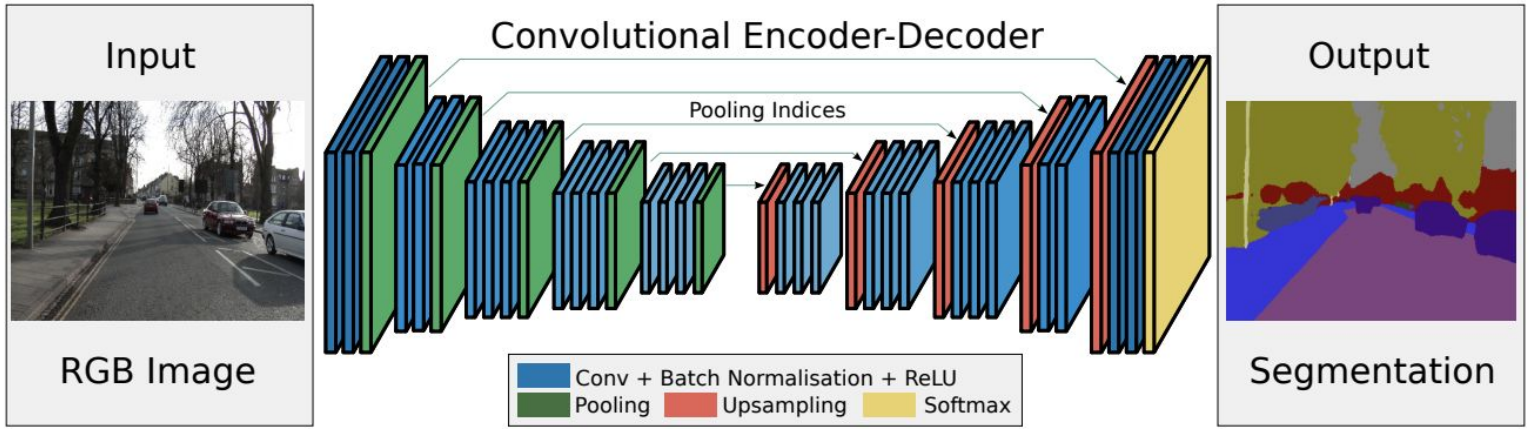




Slide credit:
https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html
https://www.freecodecamp.org/news/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050/

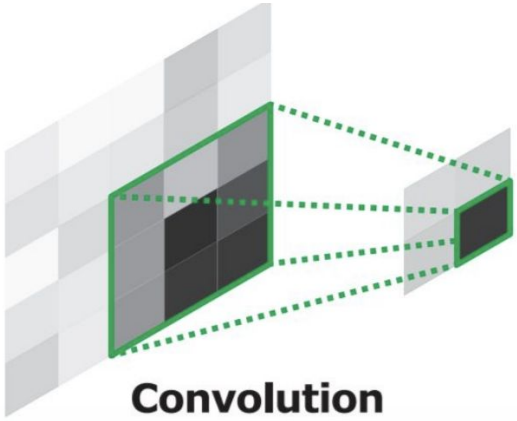
Pooling and unpooling



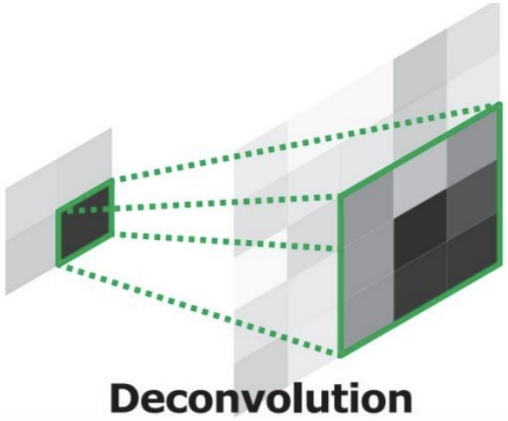


Badrinarayanan et al, SegNet (2016)

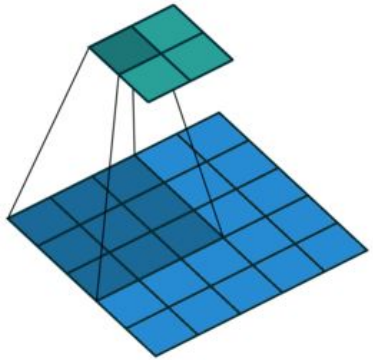
Strided Convolutions and Deconvolutions

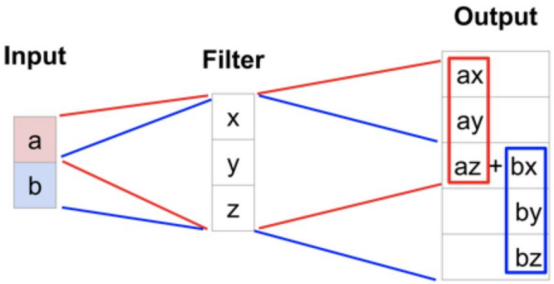


Convolution



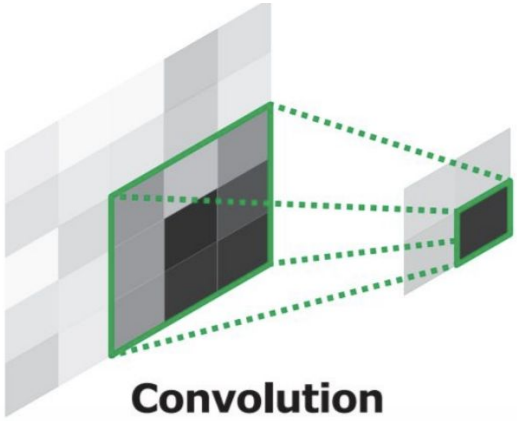
Deconvolution

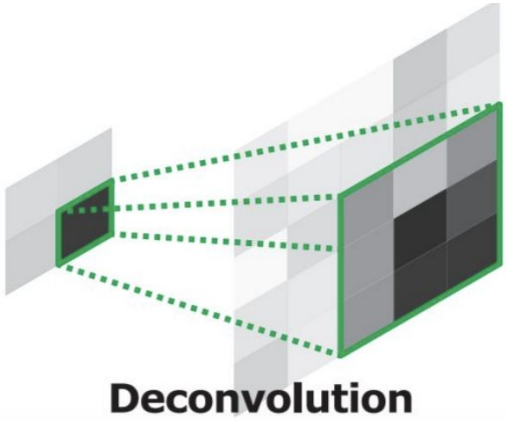


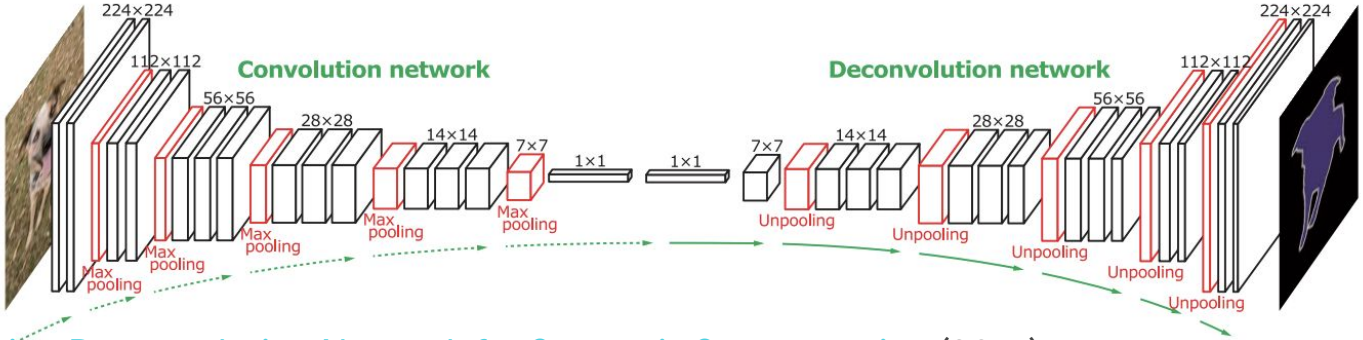


Noh et al, Learning Deconvolution Network for Semantic Segmentation (2015)

Strided Convolutions and Deconvolutions







Noh et al, Learning Deconvolution Network for Semantic Segmentation (2015)

Batch Normalization

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

Algorithm 1: Batch Normalizing Transform, applied to activation x over a mini-batch.

m = mini-batch size

For convnets: normalize by samples in mini-batch and number of pixels

Learn a pair of \gamma and \beta per feature map.

No bias in preceding conv layer.

From the original paper, Ioffe and Szegedy (2015)

Initialization in convnets

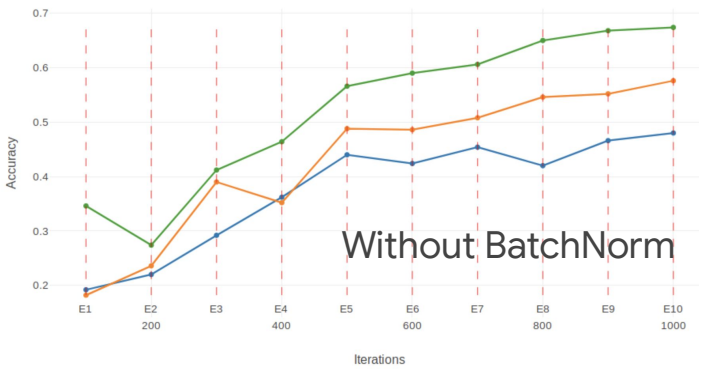
- Initialize all weights to 0?

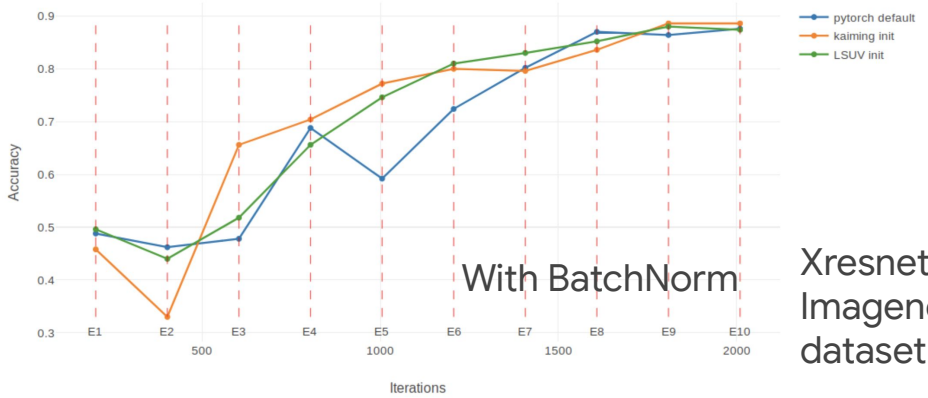
https://towardsdatascience.com/how-to-initialize-a-neural-network-27564cfb5ffc

Initialization in convnets

- Initialize all weights to 0 - symmetry issue: all weights will remain 0;
- Random init ++ for kernel weights; biases start from 0
  - Xavier initialization (2010)
  - He initialization (2015)
  - LSUV All you need is a good init (2015) - algorithmic

$$Normalized\ random\ weights \times \sqrt{\frac{2}{size\ of\ input\ vector}}$$

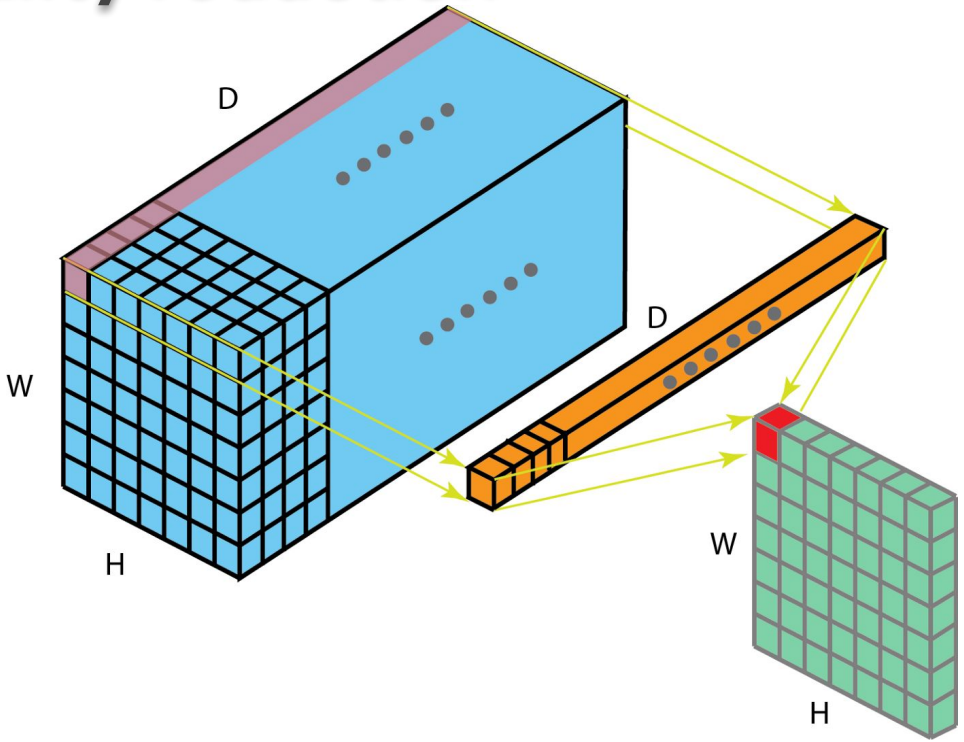




Xresnet50
Imagenette
dataset

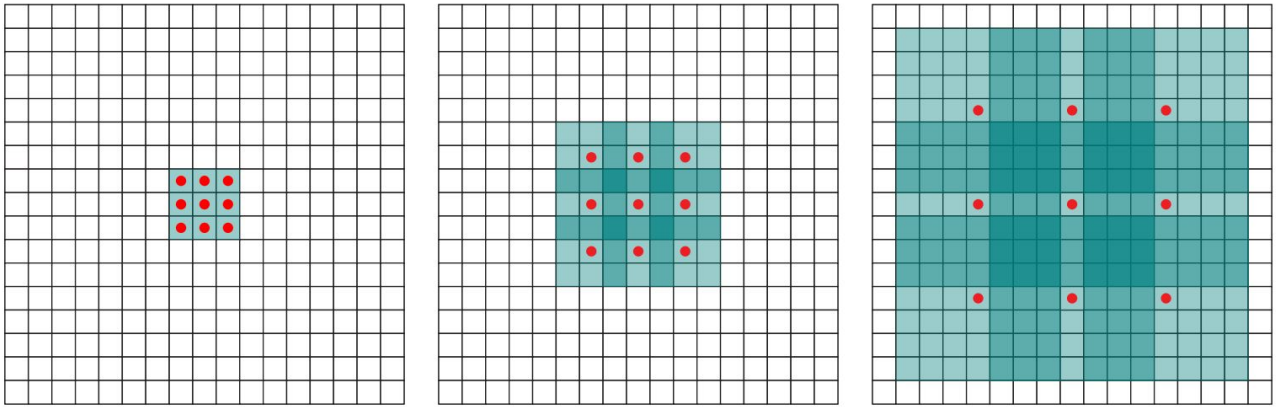
https://towardsdatascience.com/how-to-initialize-a-neural-network-27564cfb5ffc

Other convolutions: 1x1 kernels - learnt dimensionality reduction



https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215

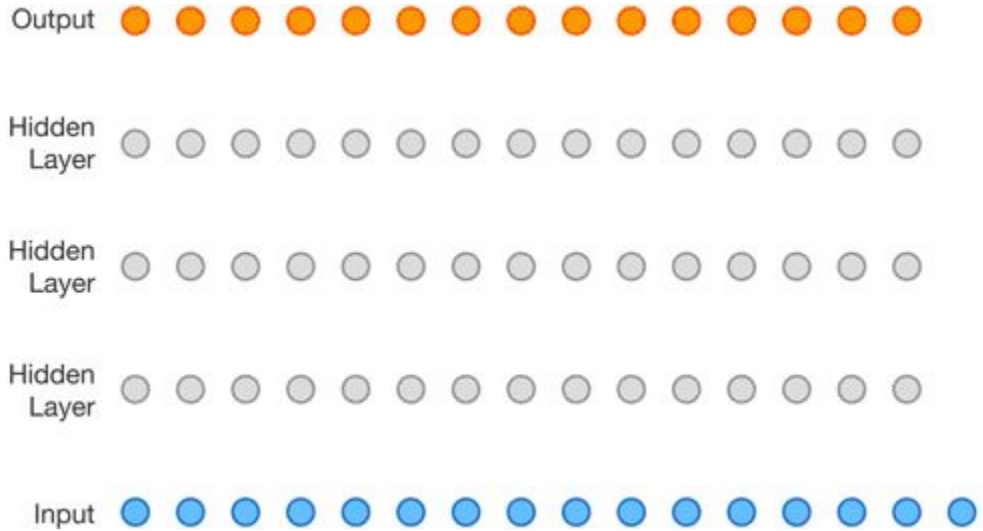
Dilated convolutions (atrous convolutions)



Effective receptive field grows exponentially while the number of parameters grows only linearly with layers.

Source: https://arxiv.org/pdf/1511.07122.pdf

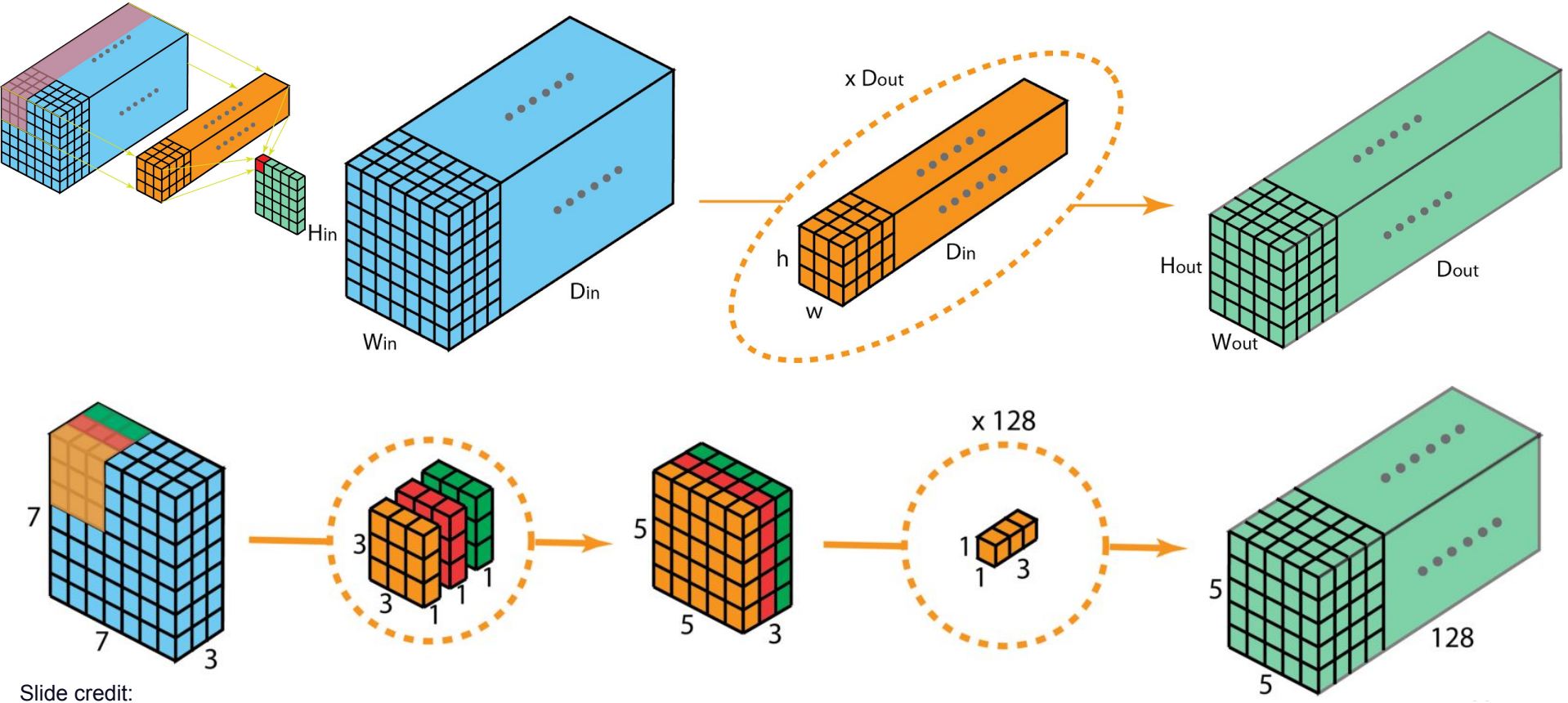
Dilated convolutions (atrous convolutions)



Effective receptive field grows exponentially while the number of parameters grows only linearly with layers.

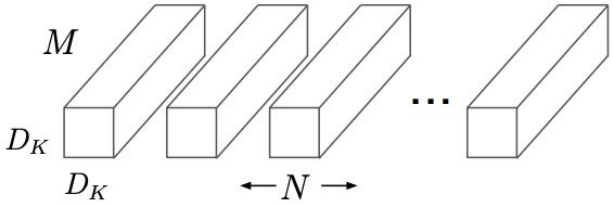
https://deepmind.com/blog/wavenet-generative-model-raw-audio/ **Wavenet architecture**

Separable convolutions

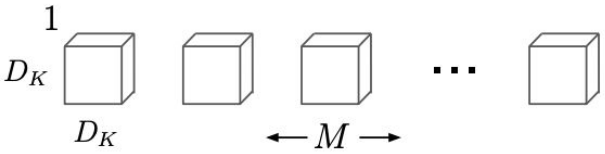


Slide credit:
Bai https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215

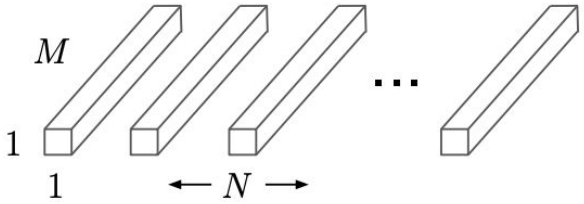
Separable convolutions: MobileNets



(a) Standard Convolution Filters



(b) Depthwise Convolutional Filters



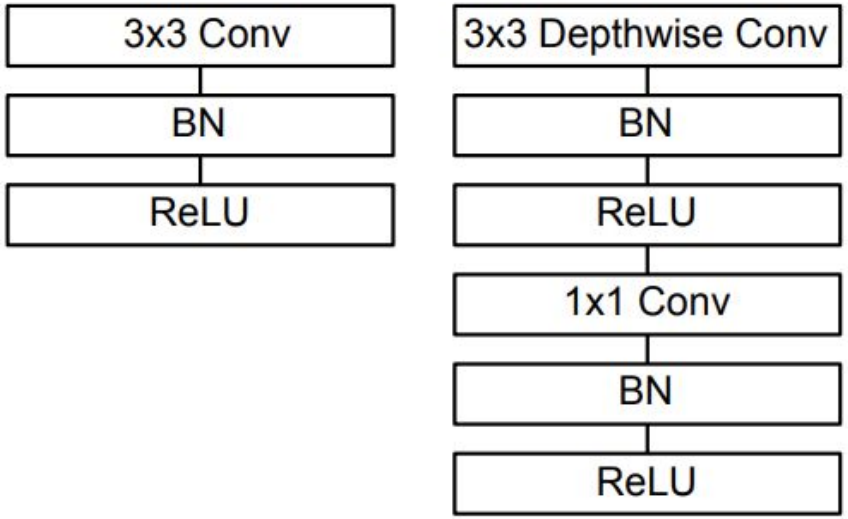
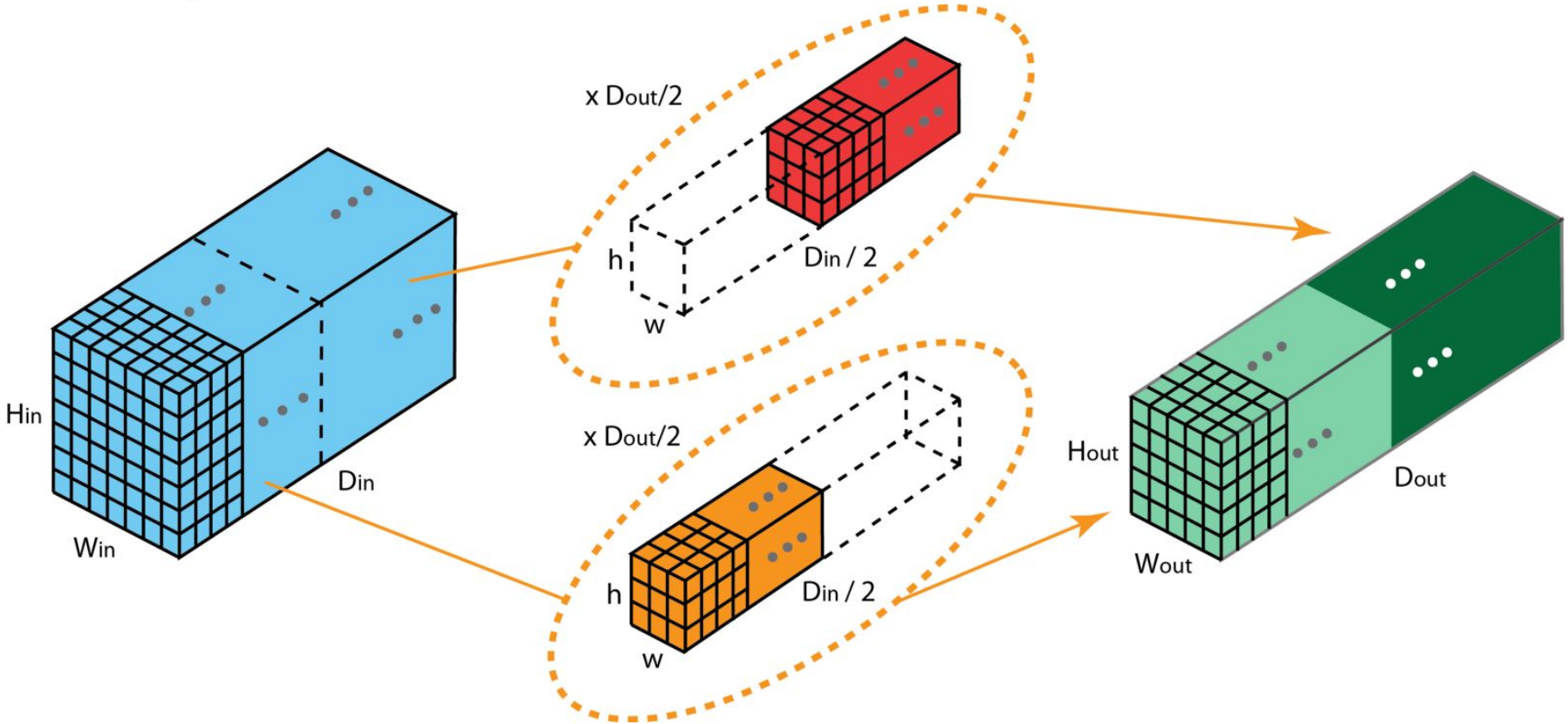


Figure 3. Left: Standard convolutional layer with batchnorm and ReLU. Right: Depthwise Separable convolutions with Depthwise and Pointwise layers followed by batchnorm and ReLU.

MobileNets, Howard et al. (2017)

Grouped convolutions



Slide credit:
Bai https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215

Dynamic lightweight convolutions

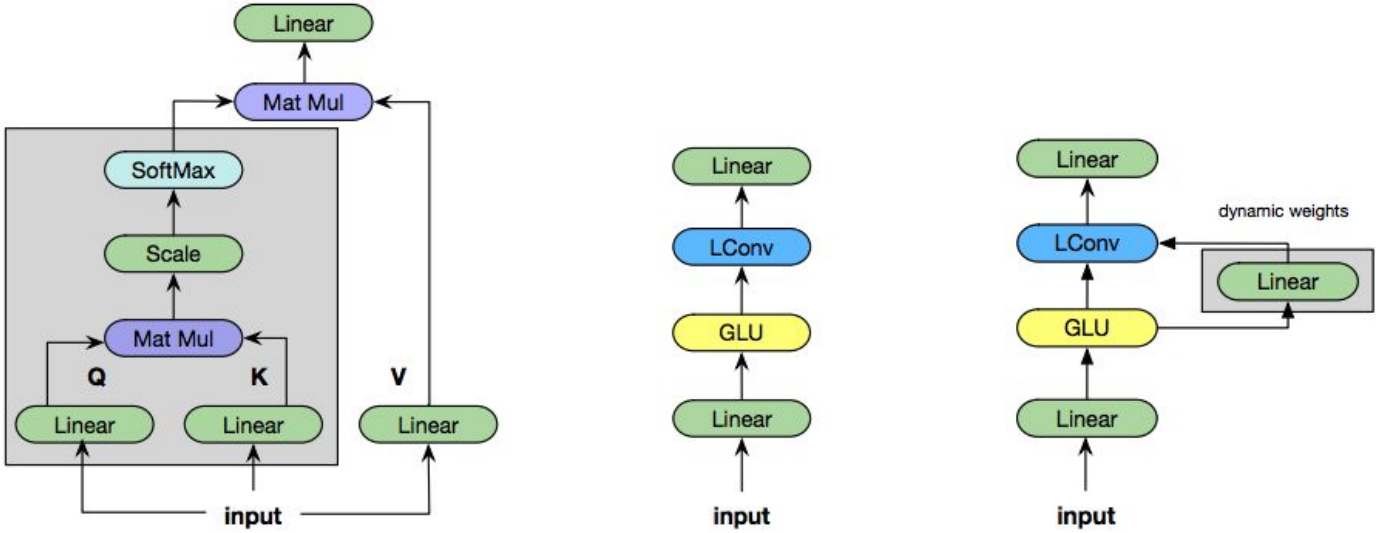


Figure 2: Illustration of self-attention, lightweight convolutions and dynamic convolutions.

[Pay less attention](), Wu et al. (2019)

Summary

- Use strided conv instead of pooling when possible
- Increase the number of feature maps when reducing spatial resolution
- Use 1x1 convolutions for dimensionality reduction
- BatchNorm important for faster and more stable training
- Initialization very important when not using BatchNorm
- Separable convolutions: good compromise for lightweight models
- A guide to convolution arithmetic for deep learning, Dumoulin and Visin (2018)

Tasks, Models, Datasets

Image classification

LeNet-5 (LeCun '98)

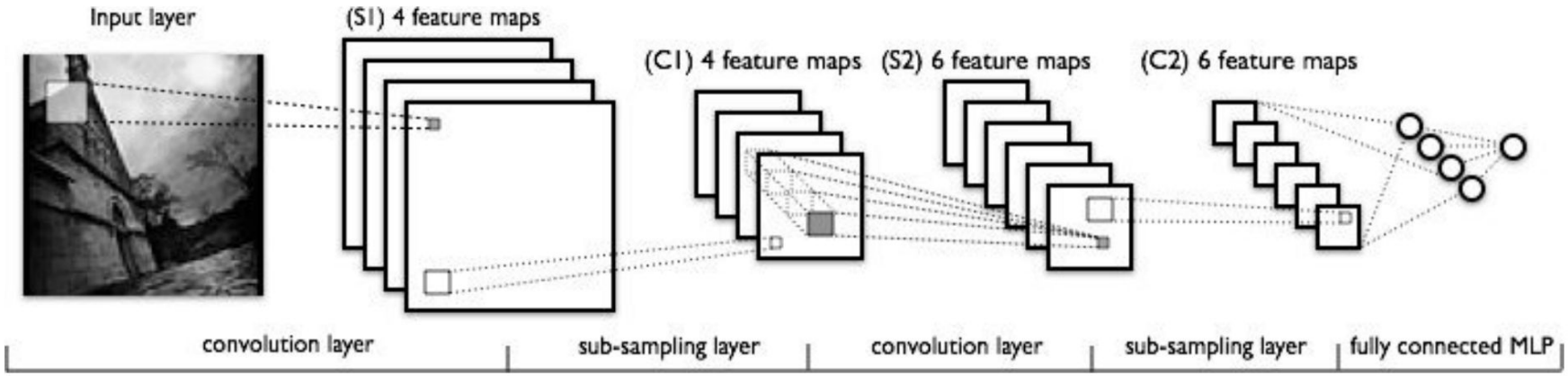
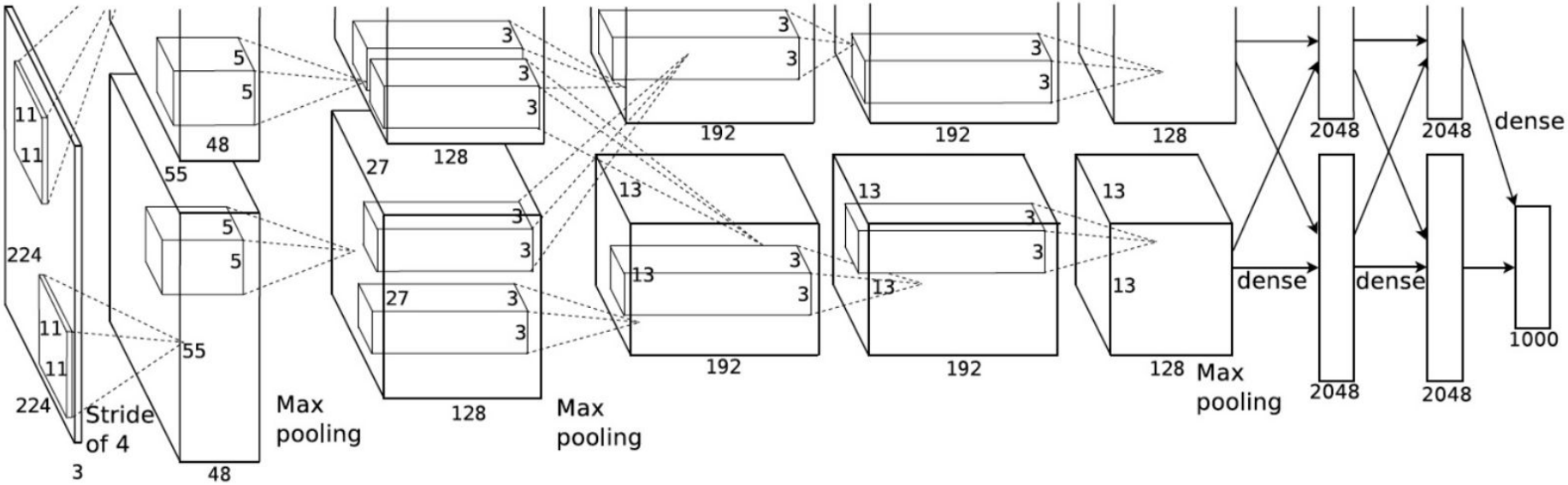


Image classification

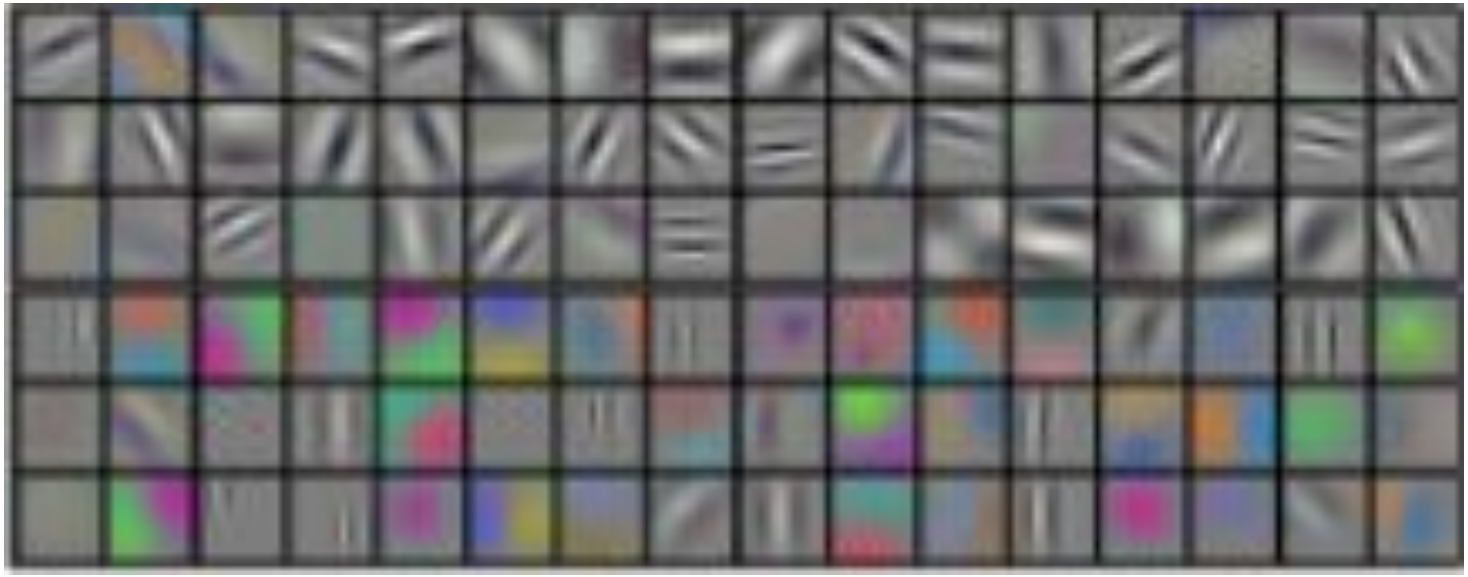
AlexNet (2012), Krishevsky et al.: Grouped convolutions



[Krishevsky et al., 2012]

Image classification

AlexNet (2012), Krishevsky et al.: Grouped convolutions



[Krishevsky et al., 2012]

Image classification

VGG, Simonyan et al (2015), 138 million parameters, only 3x3 filters, many of them

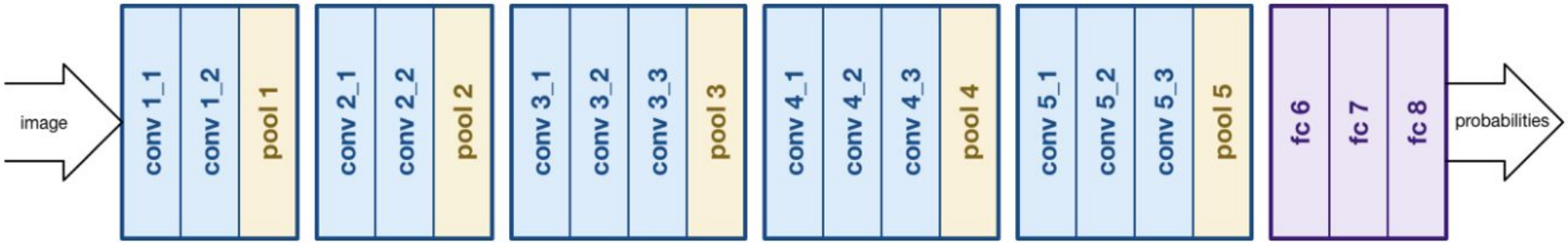


Image classification

Inception, Szegedy et al (2014), Going deeper with convolutions

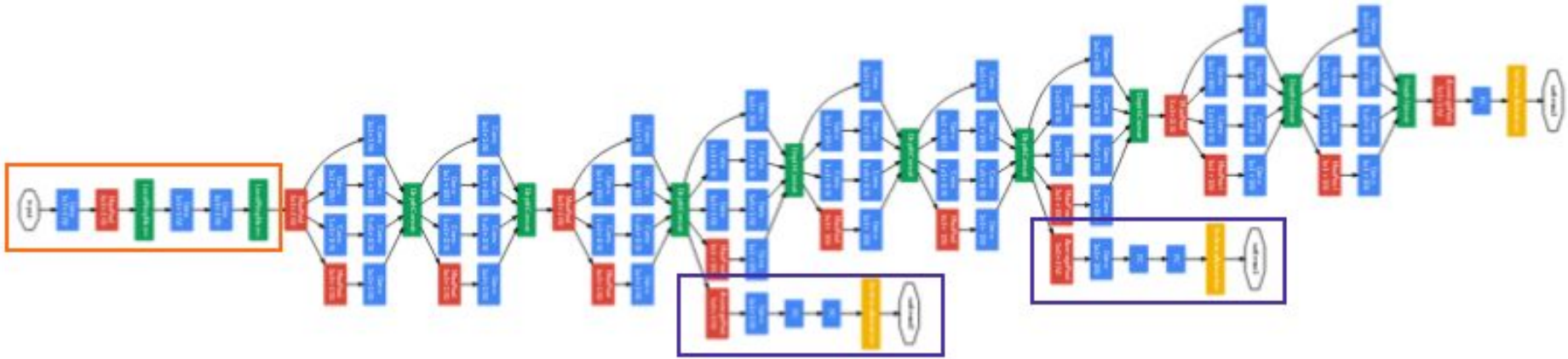


Image classification

Inception module, Szegedy et al (2014), Going deeper with convolutions

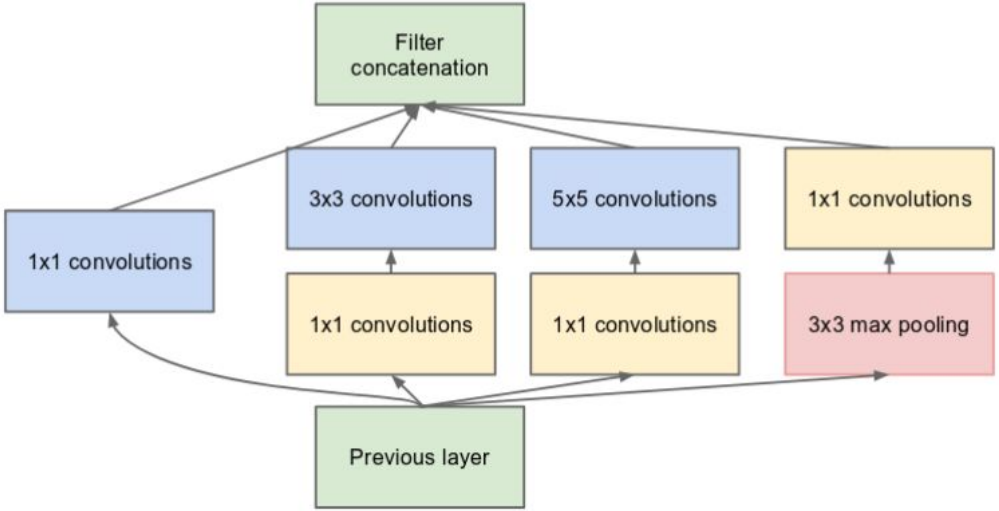


Image classification

Resnets, He et al (2015), [Deep Residual Learning for Image Recognition]

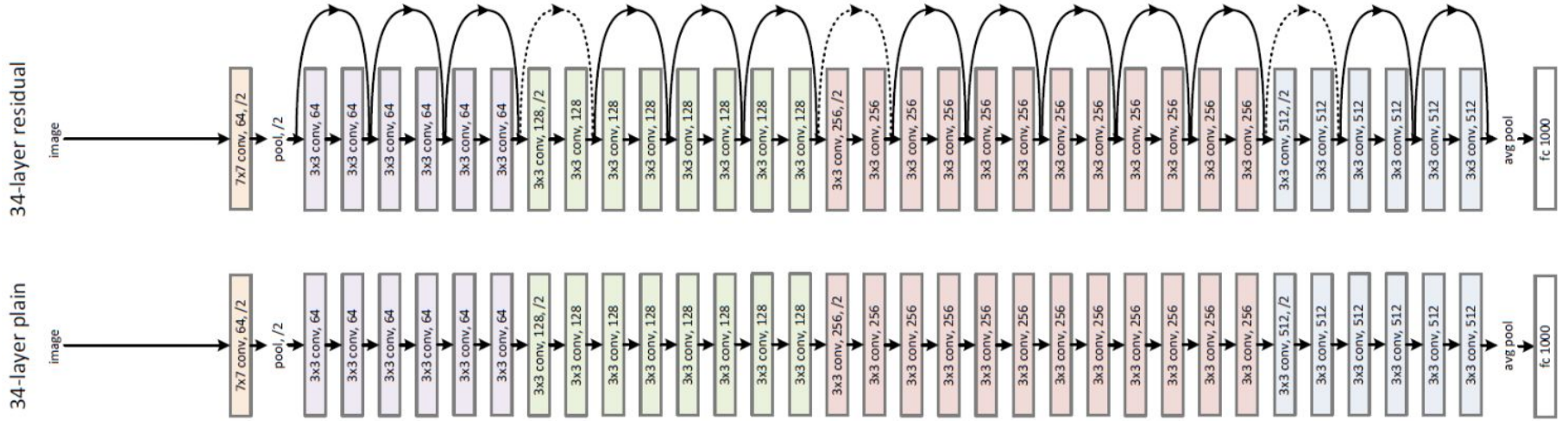


Image classification

Resnets, He et al (2015), Deep Residual Learning for Image Recognition

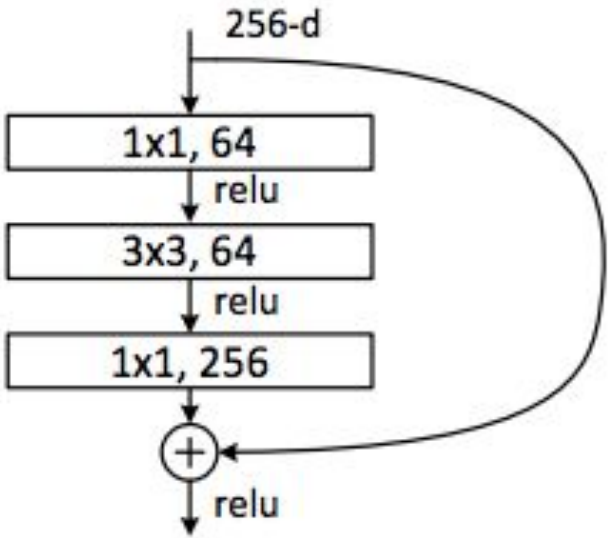
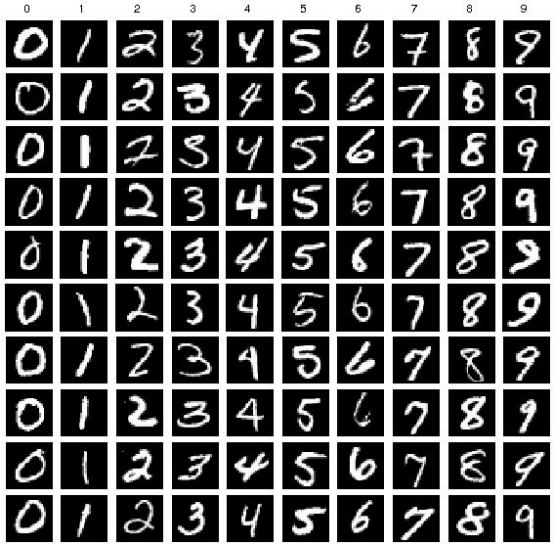
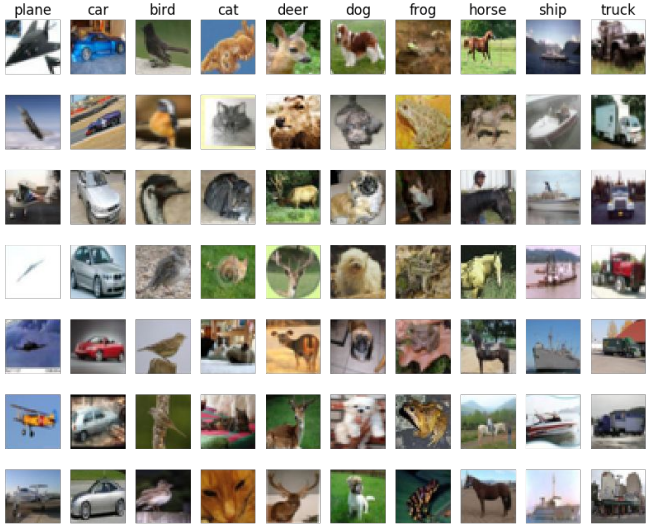


Image classification datasets

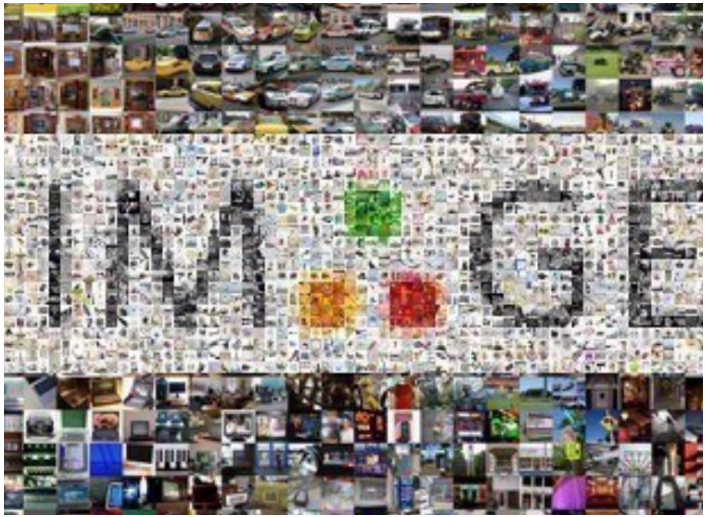
MNIST: 10 classes
50k train, 10k test



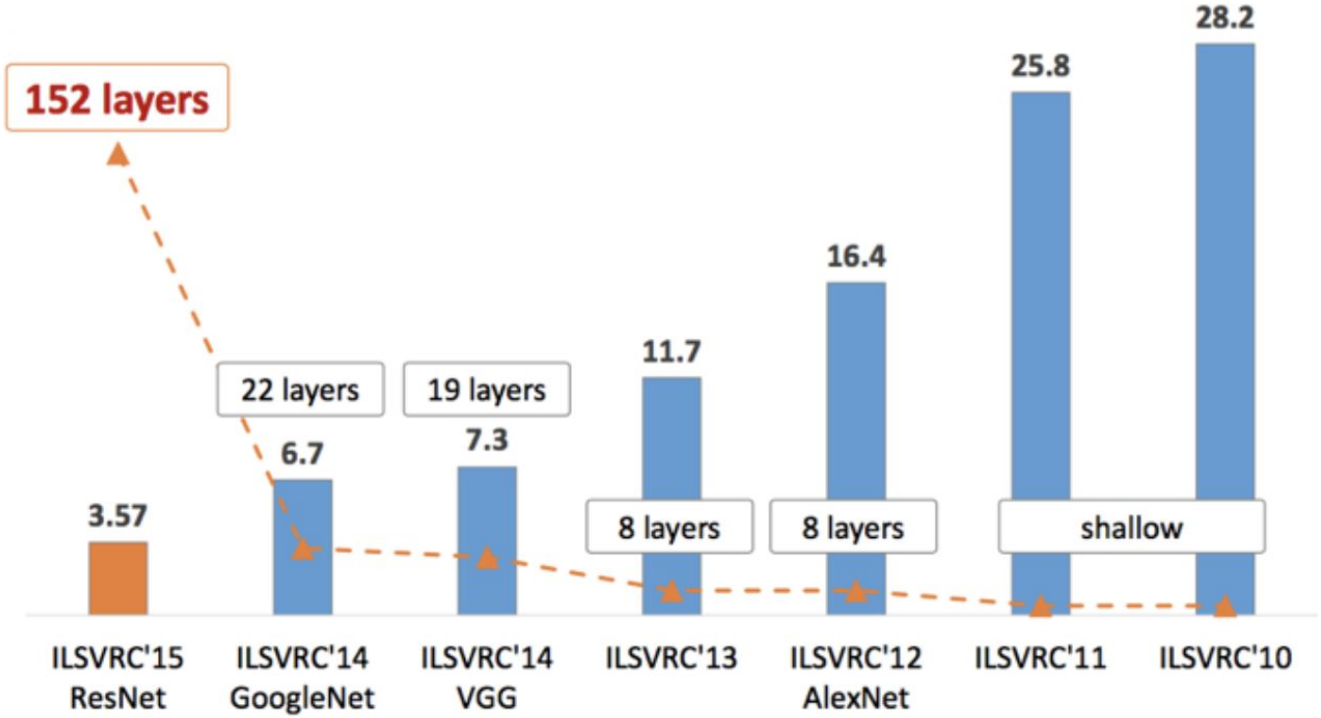
Cifar10: 10 classes
50k train, 10k test



Imagenet: 1000 classes
1M train, 100k test



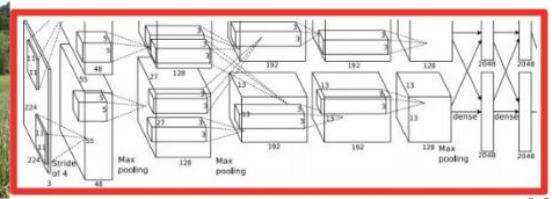
Imagenet Large Scale Visual Recognition Challenge (ILSVRC)



Object detection

Classification + Localization

This image is CC0 public domain



Often pretrained on ImageNet
(Transfer learning)

Fully Connected: 4096 to 1000

Class Scores
Cat: 0.9
Dog: 0.05
Car: 0.01
...

Correct label:
Cat

Softmax Loss

Vector: 4096

Fully Connected: 4096 to 4

Box Coordinates
(x, y, w, h)

L2 Loss

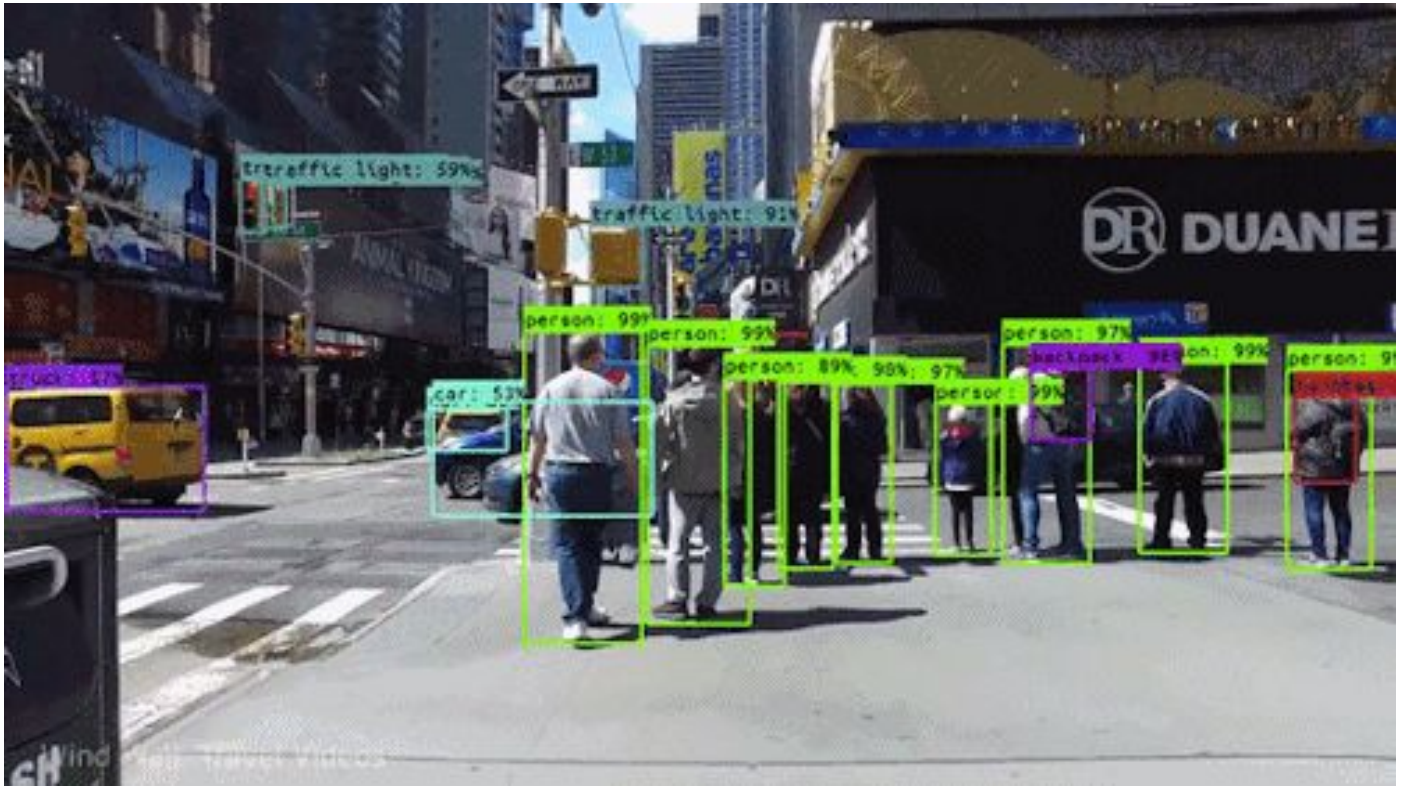
Correct box:
(x', y', w', h')

Loss

Treat localization as a regression problem!

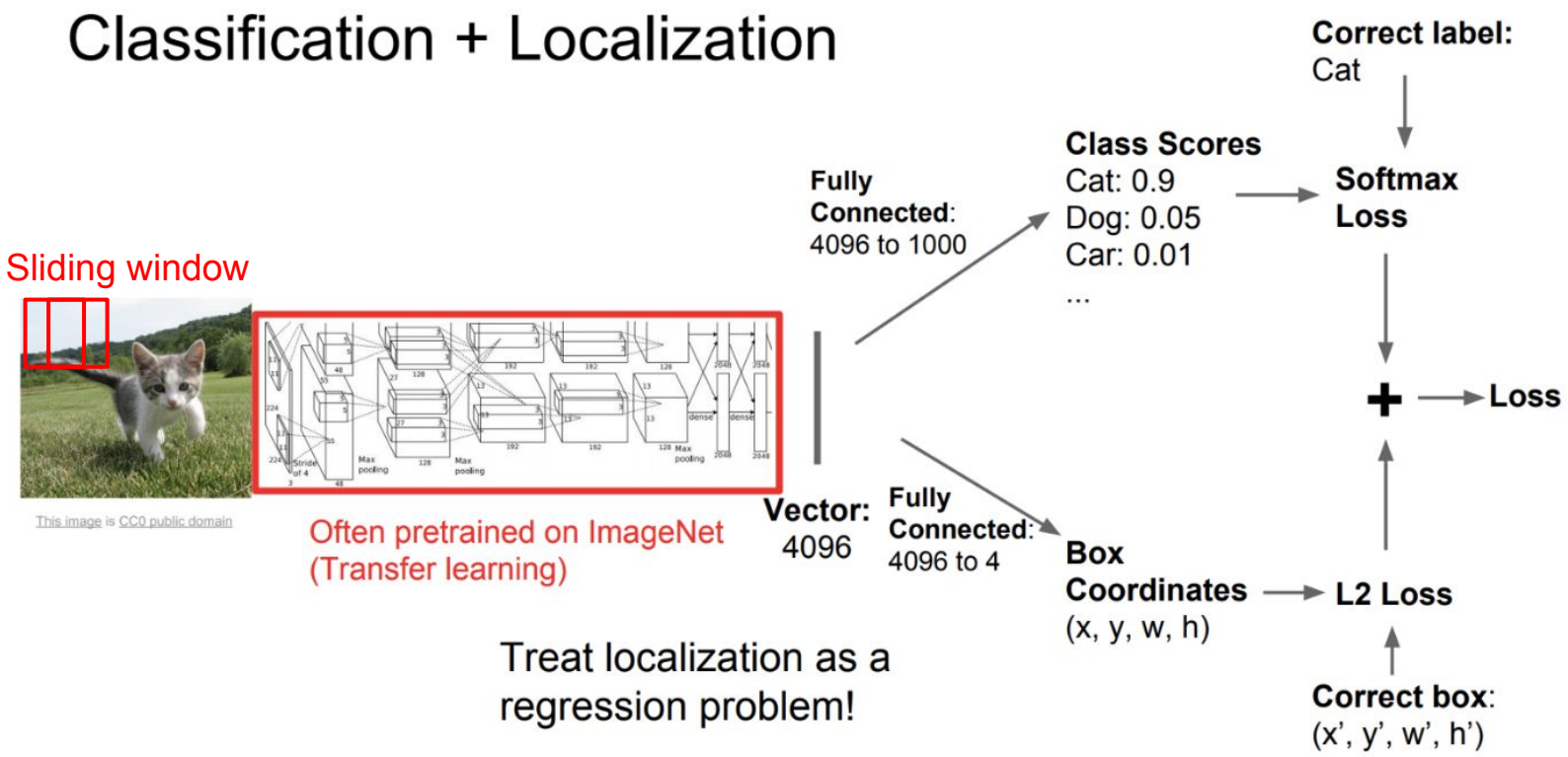
Does this work?

Object detection: multiple objects



https://towardsdatascience.com/object-detection-using-deep-learning-approaches-an-end-to-end-theoretical-perspective-4ca27eee8a9a

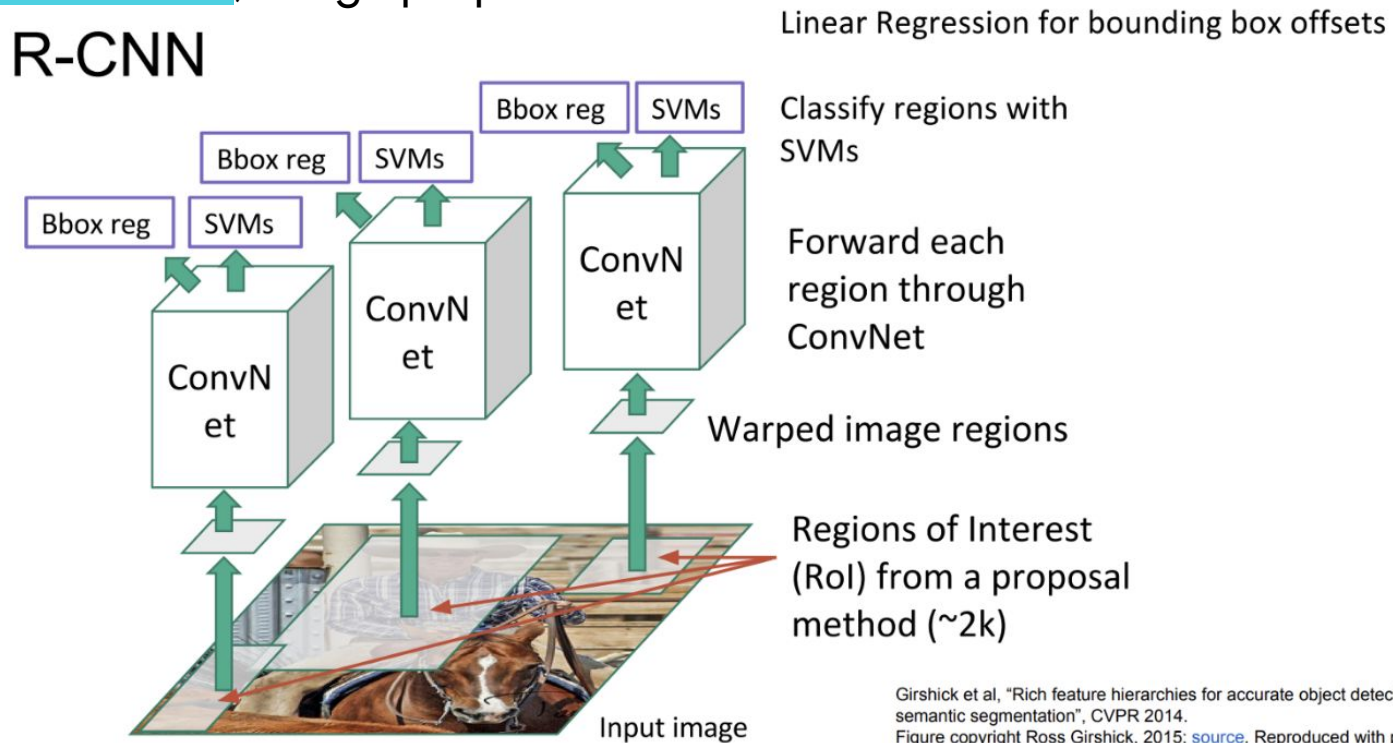
Object detection: multiple objects



https://towardsdatascience.com/object-detection-using-deep-learning-approaches-an-end-to-end-theoretical-perspective-4ca27eee8a9a

Object detection: multiple objects

Selective search, image proposals

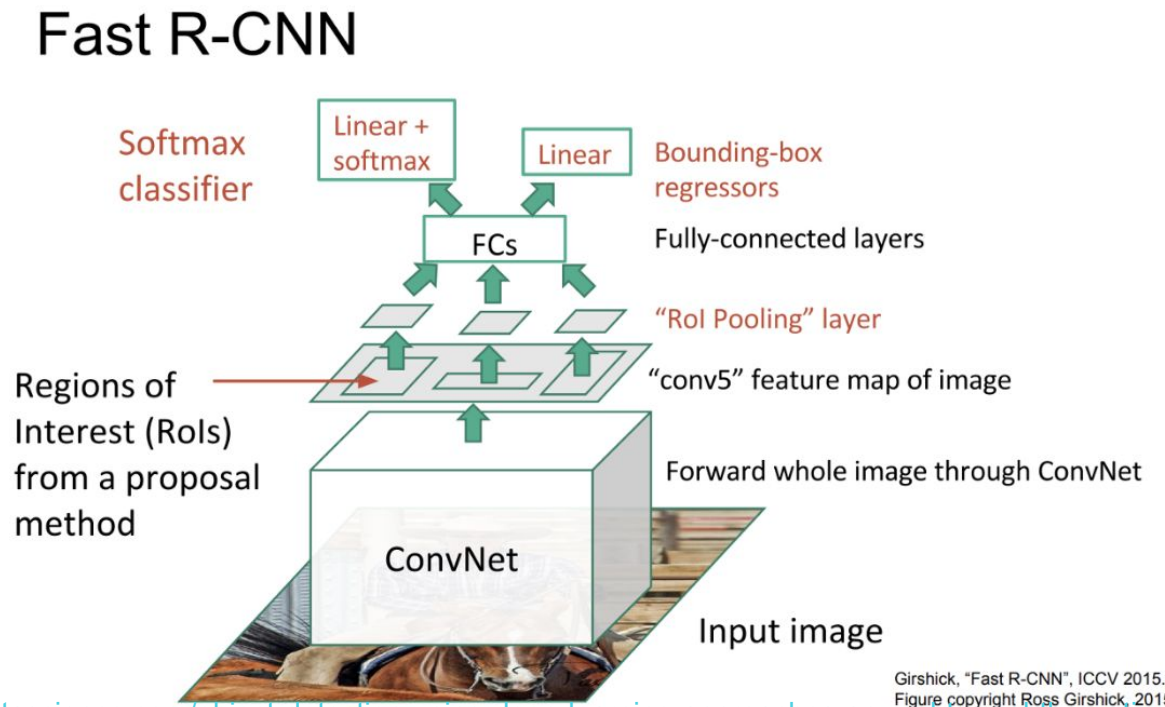


Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

https://towardsdatascience.com/object-detection-using-deep-learning-approaches-an-end-to-end-theoretical-perspective-4ca27eee8a9a

Object detection: multiple objects

Image proposals

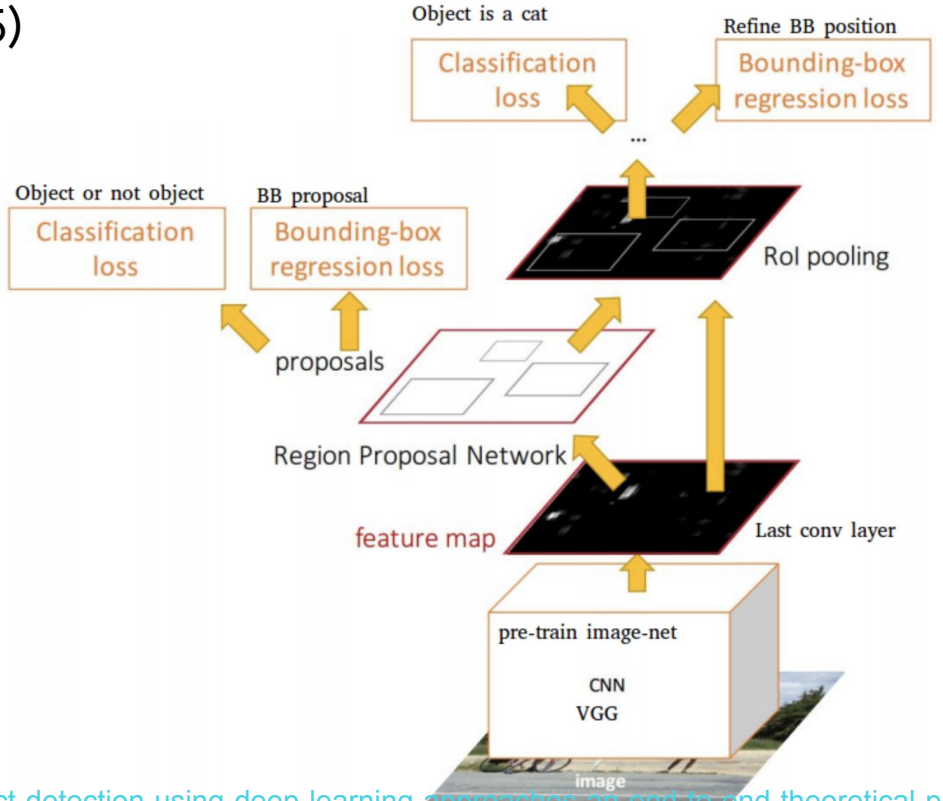


Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

https://towardsdatascience.com/object-detection-using-deep-learning-approaches-an-end-to-end-theoretical-perspective-4ca27eee8a9a

Object detection: multiple objects

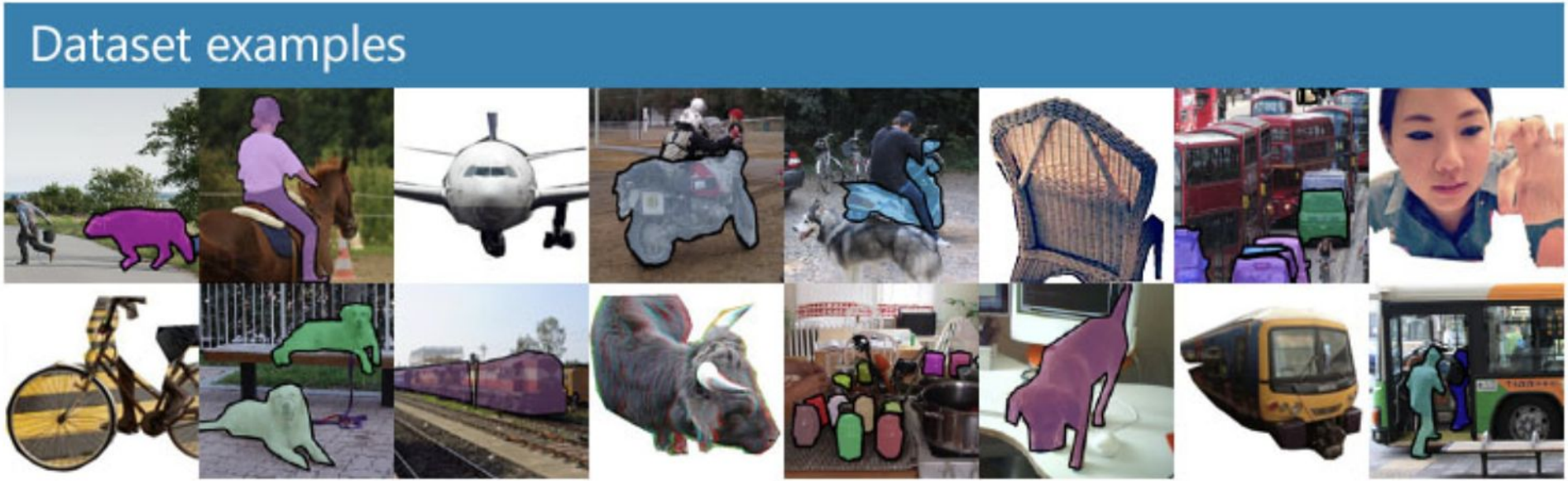
Faster-RCNN (2015)



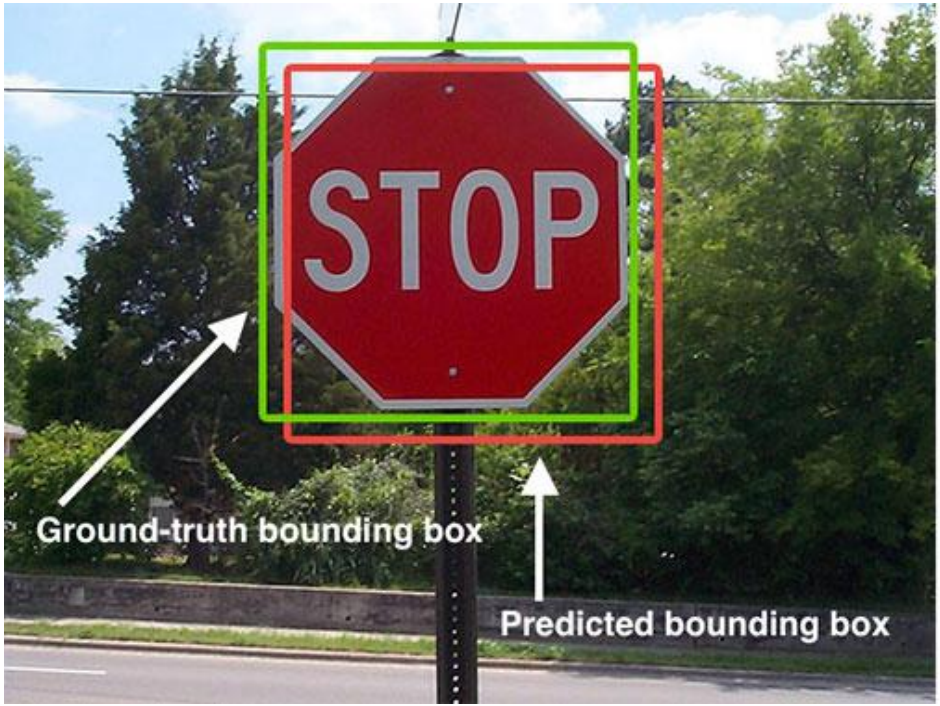
https://towardsdatascience.com/object-detection-using-deep-learning-approaches-an-end-to-end-theoretical-perspective-4ca27eee8a9a

Coco dataset

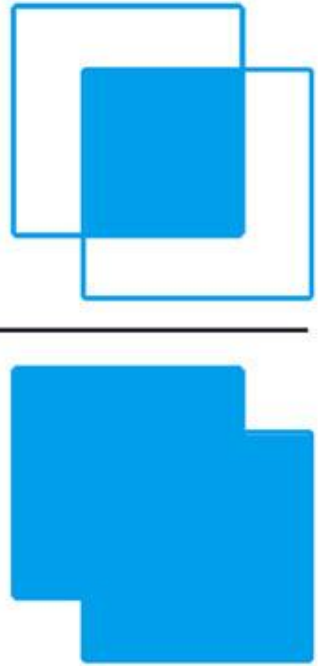
Object detection, segmentations, captions; 330k images; 80 object classes



Evaluation

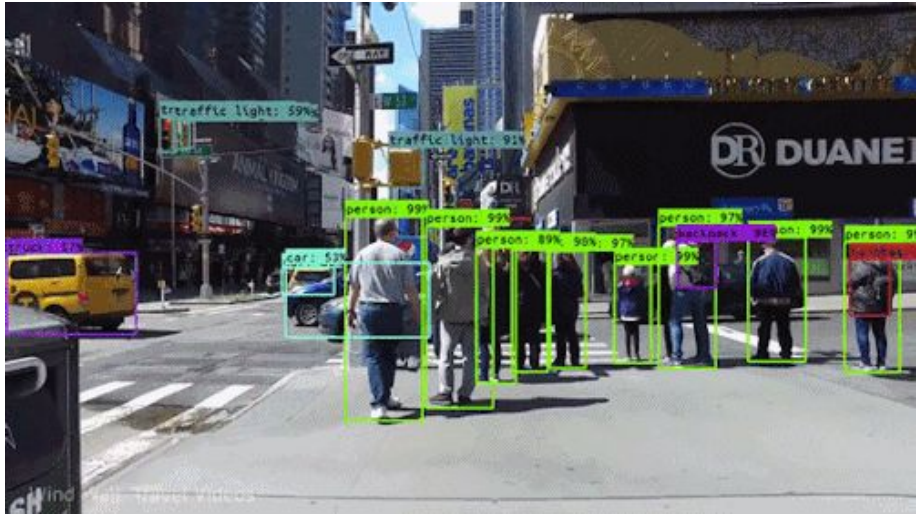


$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$



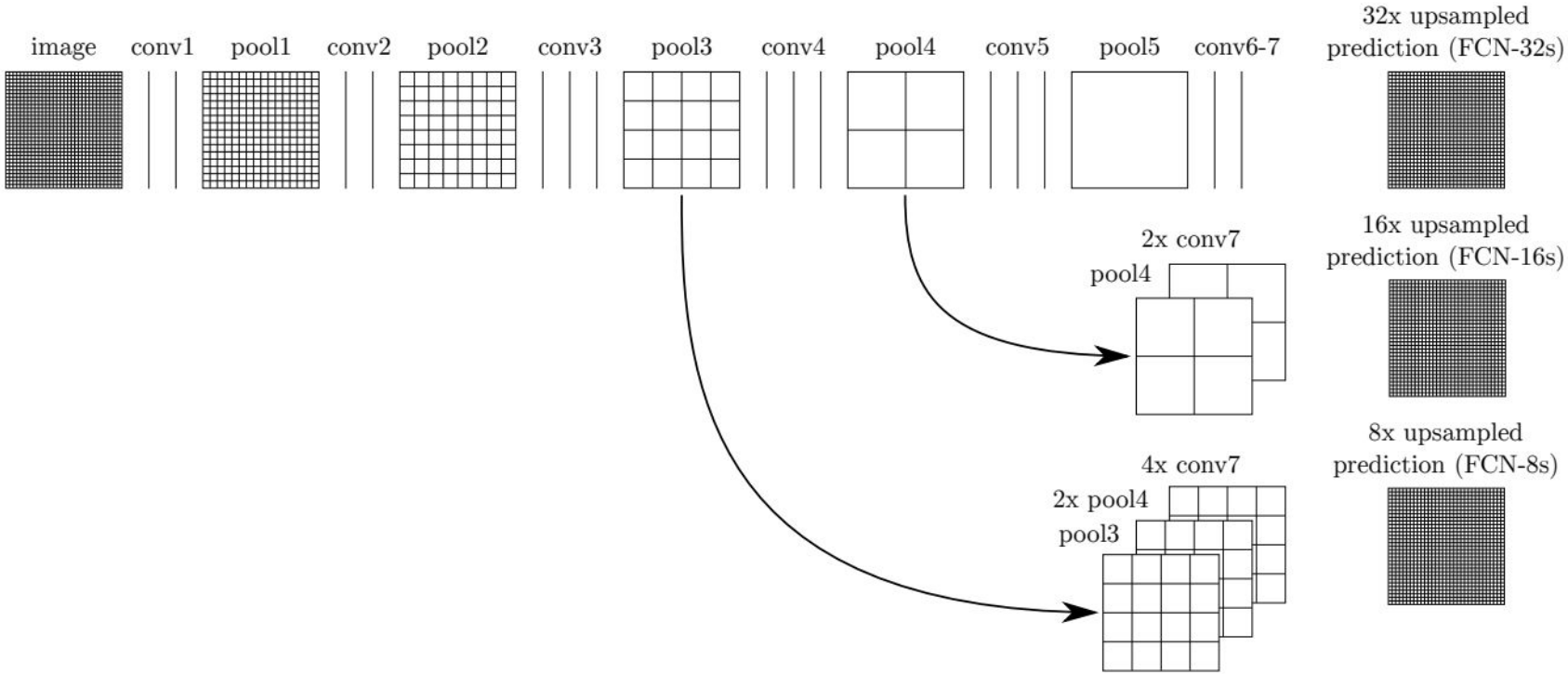
Issue?

Object detection vs semantic segmentation



Semantic segmentation

Fully Convolutional Networks for Semantic Segmentation, Long et al (2014)



Semantic segmentation dataset

Cityscapes dataset (5k annotated images) + instances + videos

Stuttgart

Zurich

Ulm

Münster

Cologne

Bonn

Summary

- Popular networks for common tasks exist; start from that.
- Don’t forget regularization (weight decay, dropout).
- Transfer learning is very powerful (Imagenet pretraining).
- Still a lot of supervision is required for finetuning.
- Many other tasks: depth estimation, pose estimation, optical flow estimation etc.

Learning representations

Triplet loss for one-shot learning

Anchor

Positive

Anchor

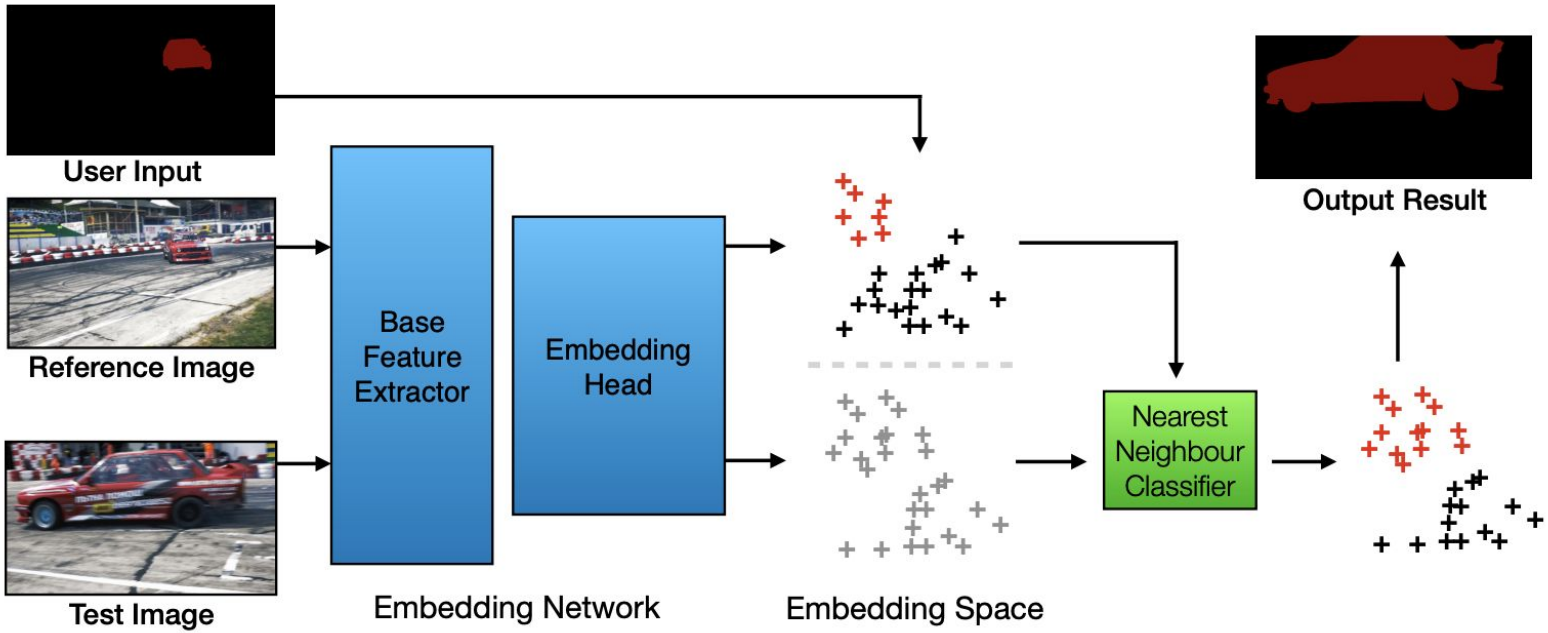
Negative

$$\mathcal{L} = max(d(a,p) - d(a,n) + margin, 0)$$

https://towardsdatascience.com/siamese-network-triplet-loss-b4ca82c1aec8

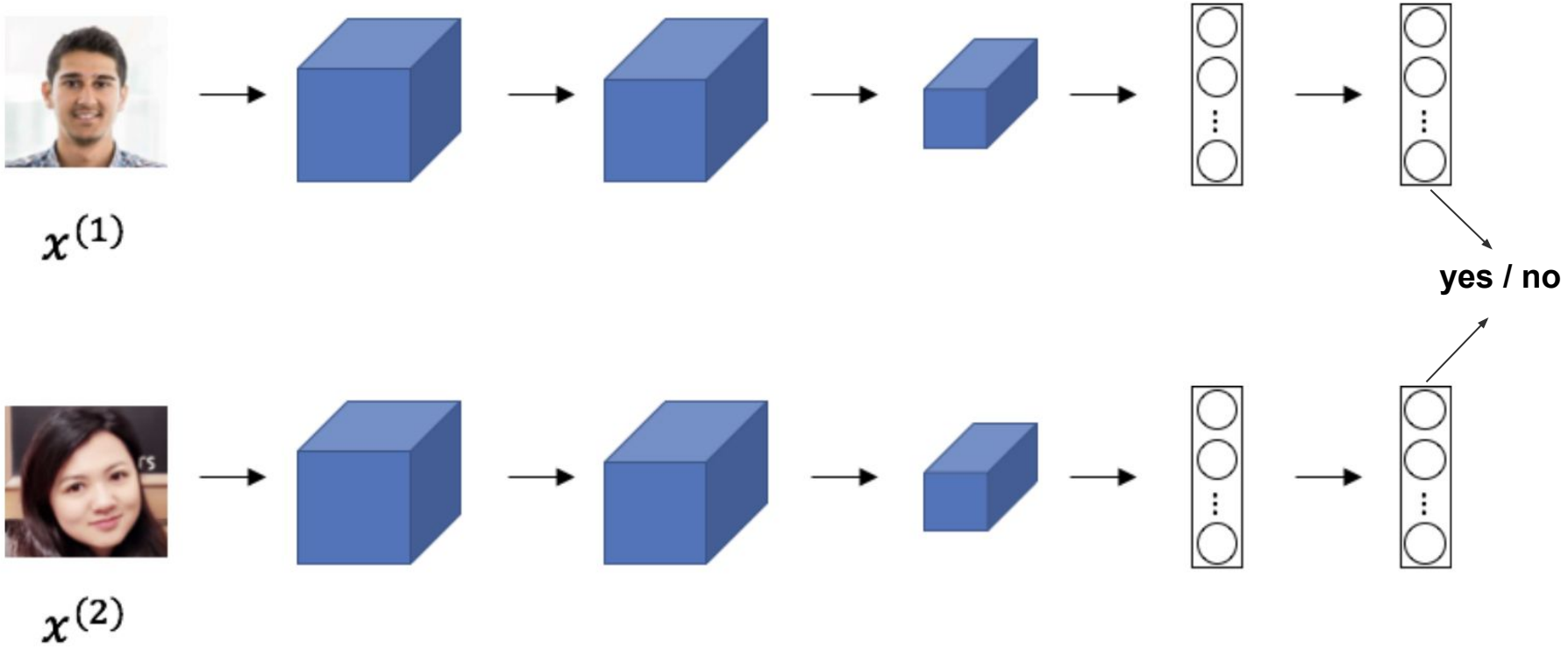
Triplet loss for instance segmentation

Blazingly Fast Video Object Segmentation with Pixel-Wise Metric Learning, Chen et al (2018)

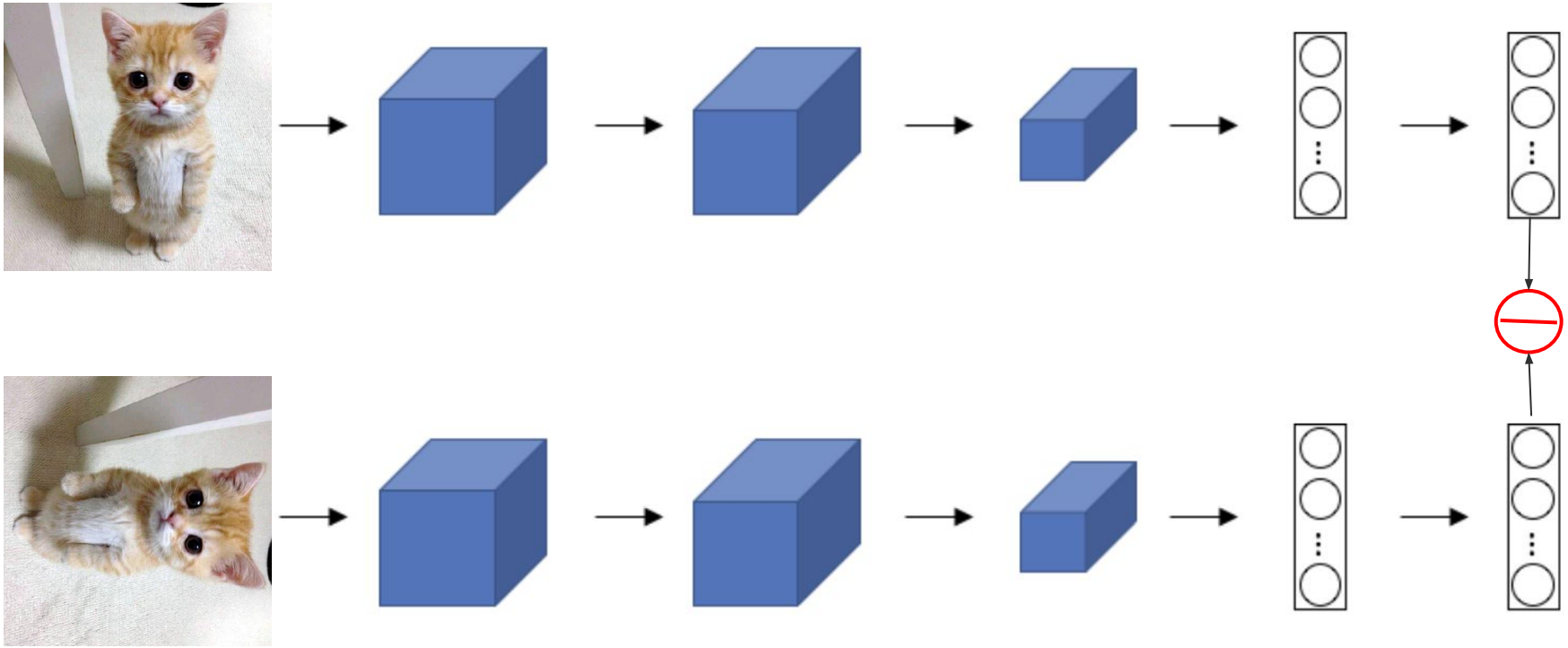


Should use videos!

Siamese networks



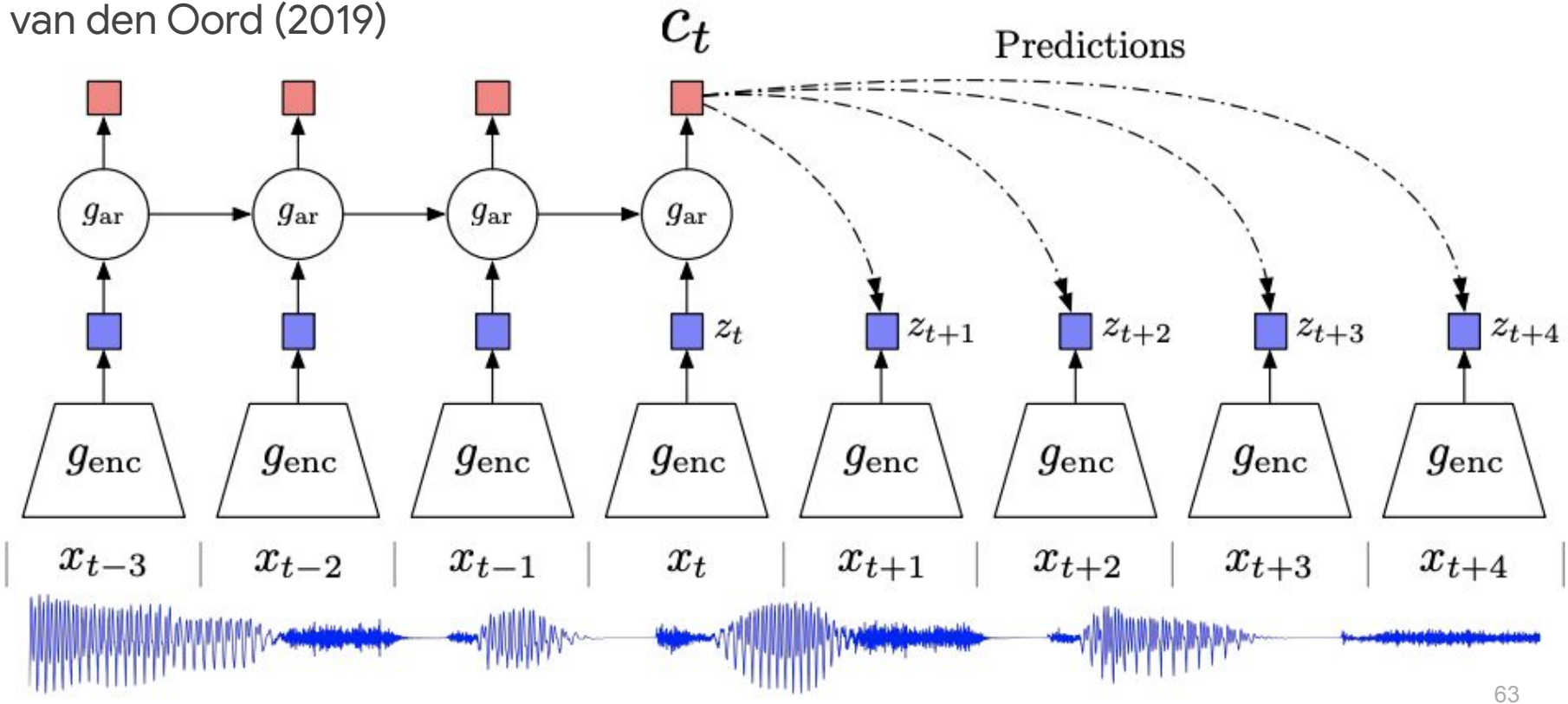
Siamese networks for invariant representations



Outputs should be the same. **Any issue?**

Contrastive predictive coding

van den Oord (2019)



Objective functions

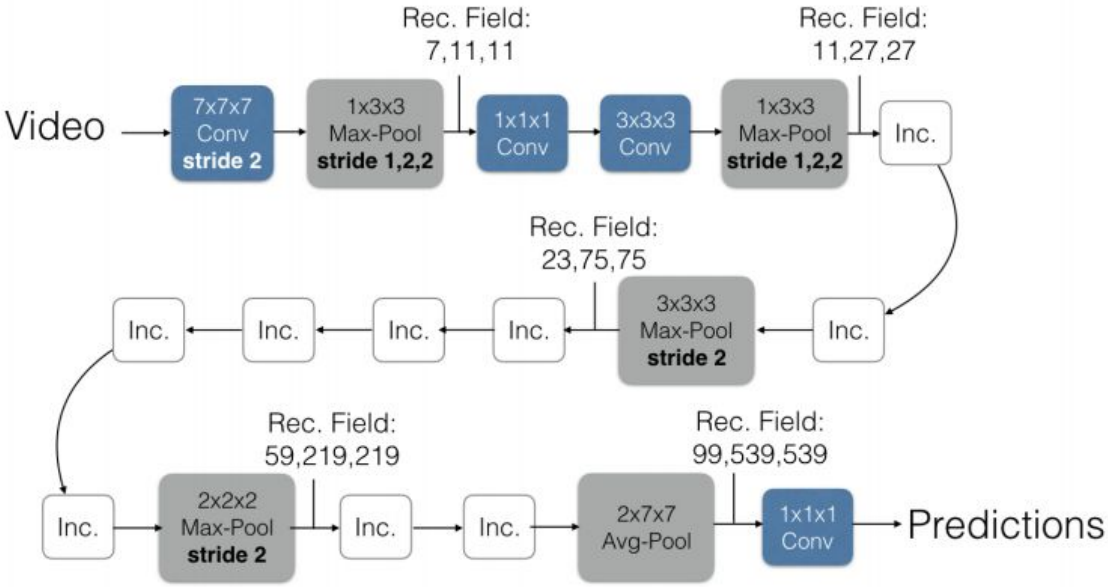
Output activation function	Loss function	Probabilistic interpretation
Linear	Mean square error	Gaussian
(binary classification) Sigmoid	Binary cross-entropy	Bernoulli
(multi-label classification) Softmax	Negative log-likelihood	Multinomial

Slide credit: R. Pascanu

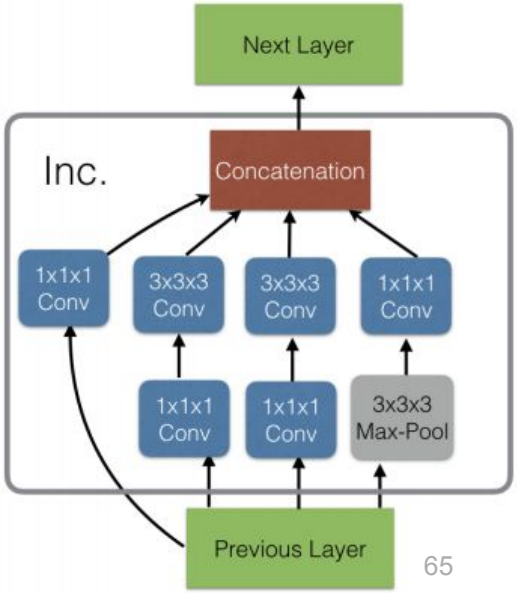
Beyond images: Videos

3D convolutions for action recognition, [Carreira and Zisserman (2017)]

Inflated Inception-V1

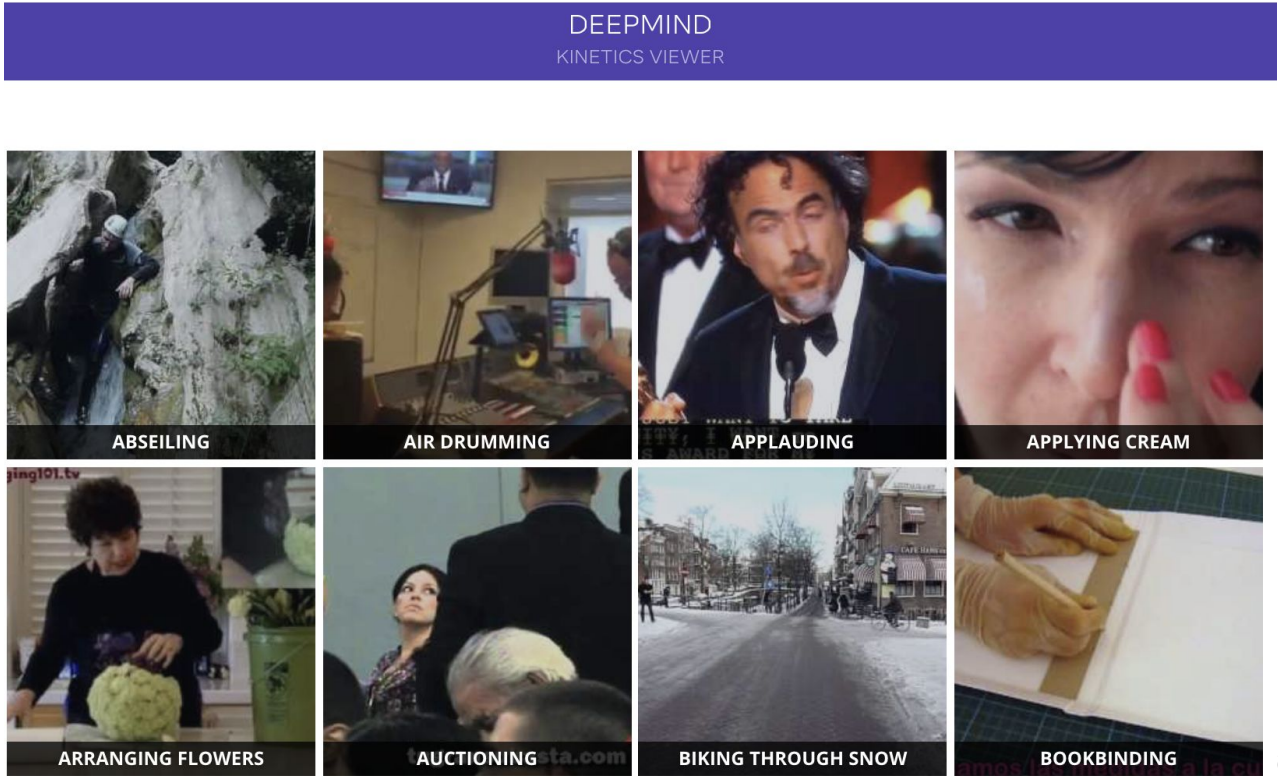


Inception Module (Inc.)



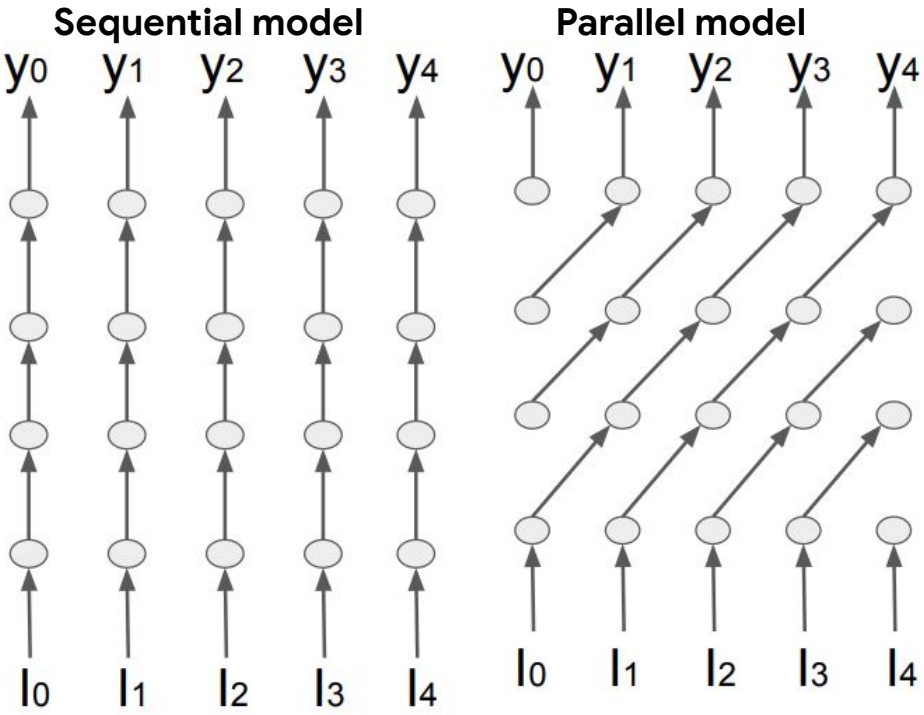
Beyond images: Videos

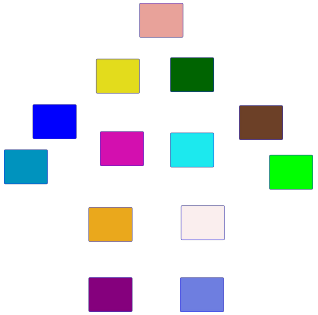
Kinetics dataset for action recognition:
700 classes,
~ 700k videos



Beyond images: Videos

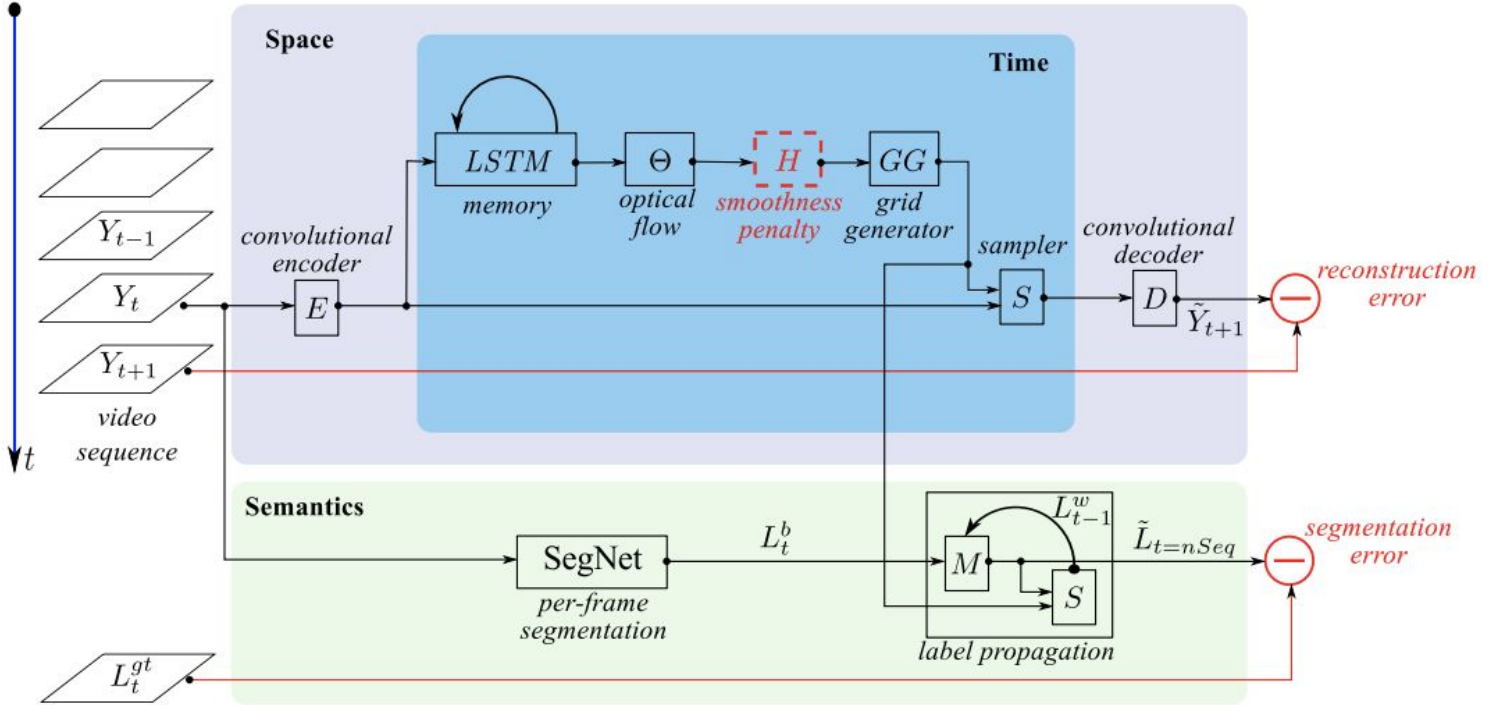
Efficient parallel video processing by breaking the sequential dependencies, [Carreira, Patraucean et al. 2018]





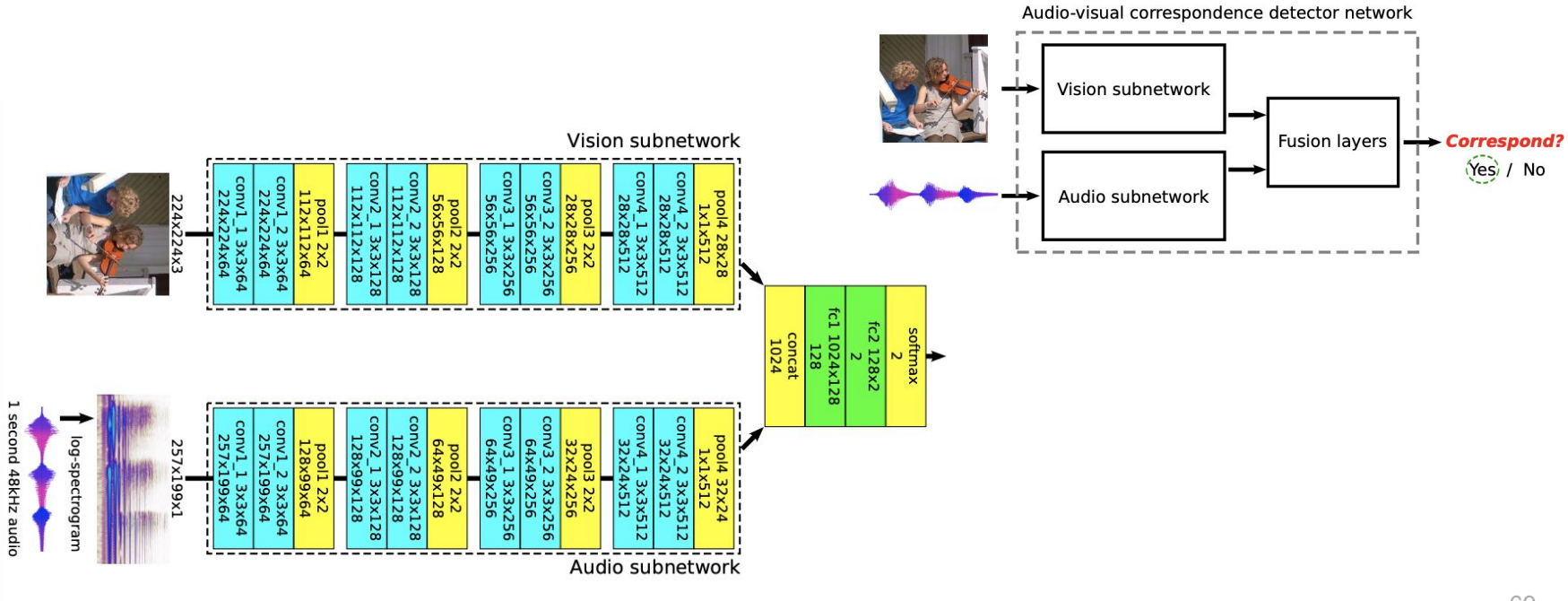
Beyond images: Videos

ConvLSTM, Patraucean et al (2015): label propagation through implicit optical flow

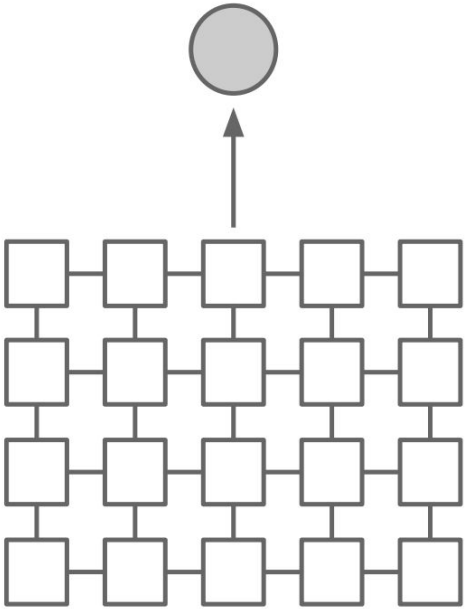


Beyond images: Sound

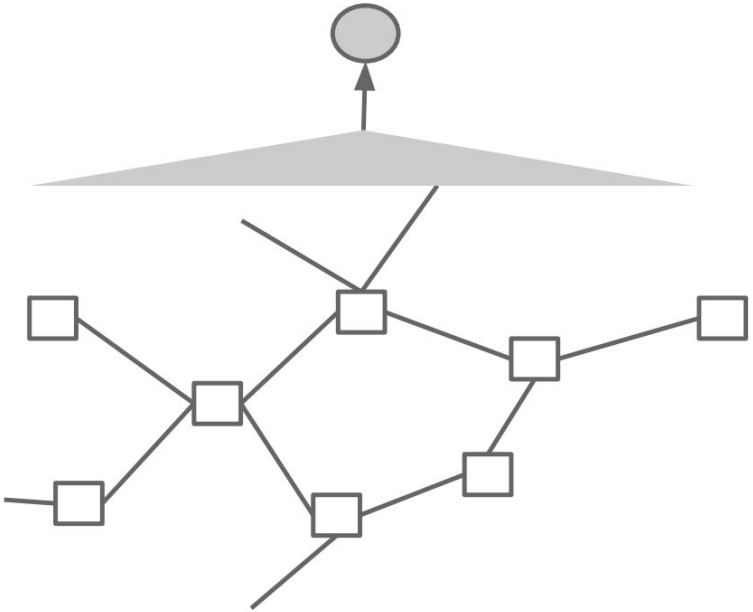
Look, listen, and learn, Arandjelovic and Zisserman (2017)



Beyond CNNs: Graph Convnets



Grid structure; spatial locality

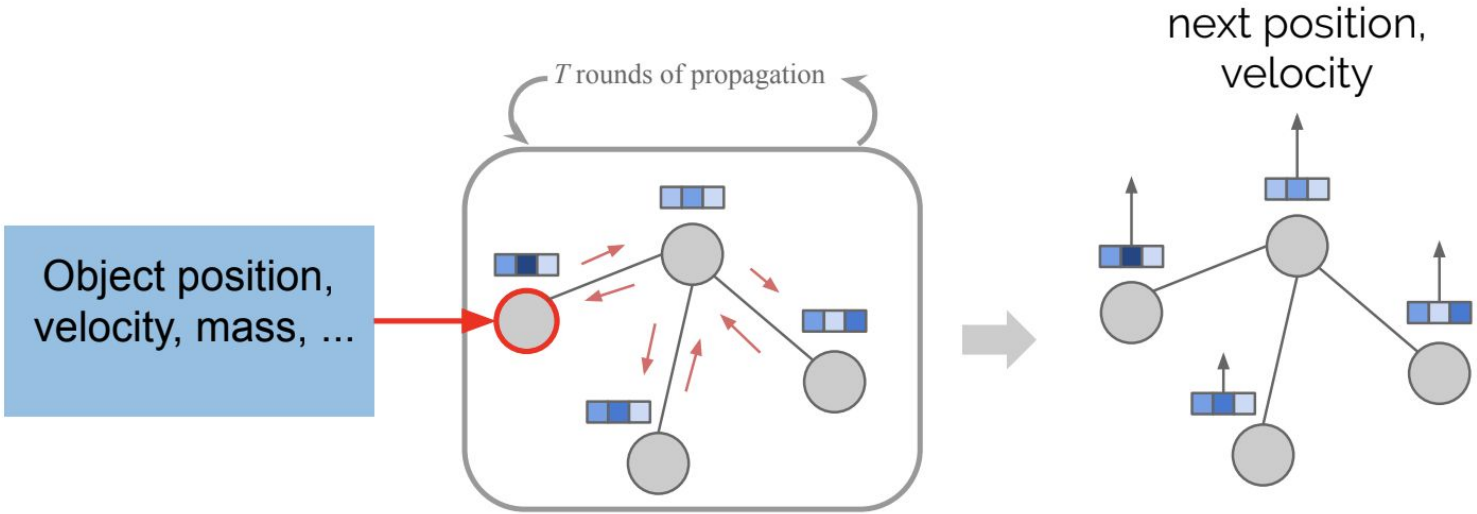


No grid; Graph structure

Slide credit: R. Pascanu

Examples of data without grid structure?

Beyond CNNs: Graph Convnets

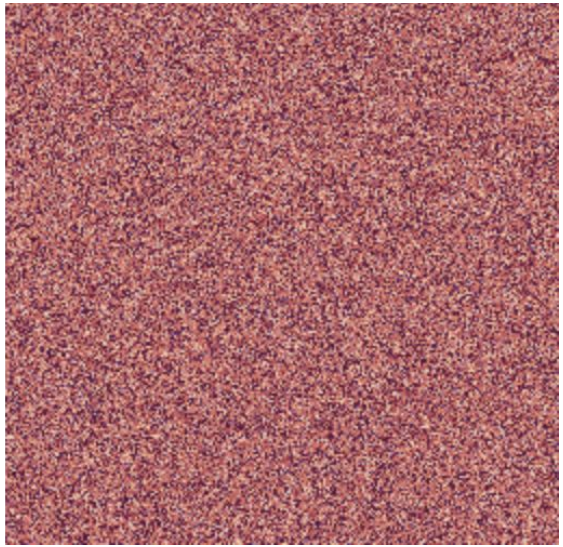


Slide credit: R. Pascanu

Beyond CNNs: Transformers

When data biases are not valid

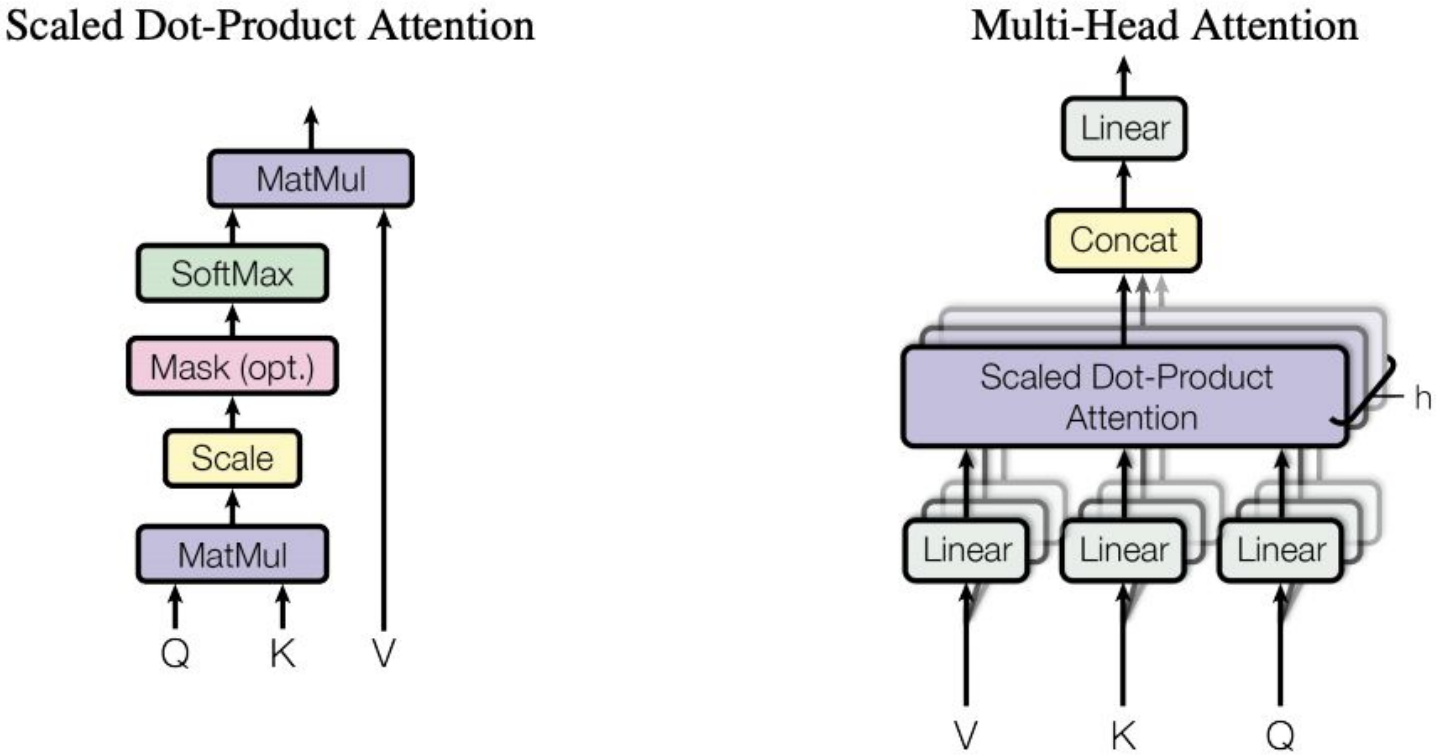
Lena image



Permuted image

Beyond CNNs: Transformers

Non-local computation; pairwise interactions



Conclusion

- Convnets are very powerful image models; maybe not general enough?
- Video models lag behind.
- Transformers: promising models applicable on any type of data; scalability issue due to quadratic complexity of pairwise interactions.
- Too much supervision required. Promising directions: self-supervision; cross-supervision in multi-modal settings.
- Computer Vision works, but far from being solved!
- My research goal: discover objects in videos without supervision.