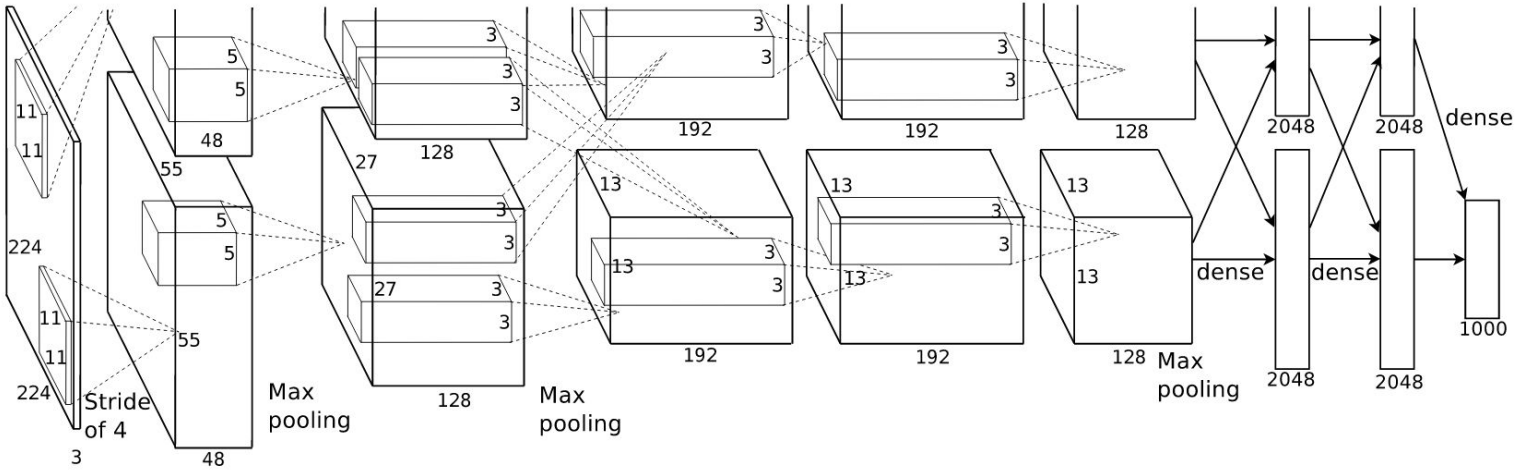
Alexnet & Beyond

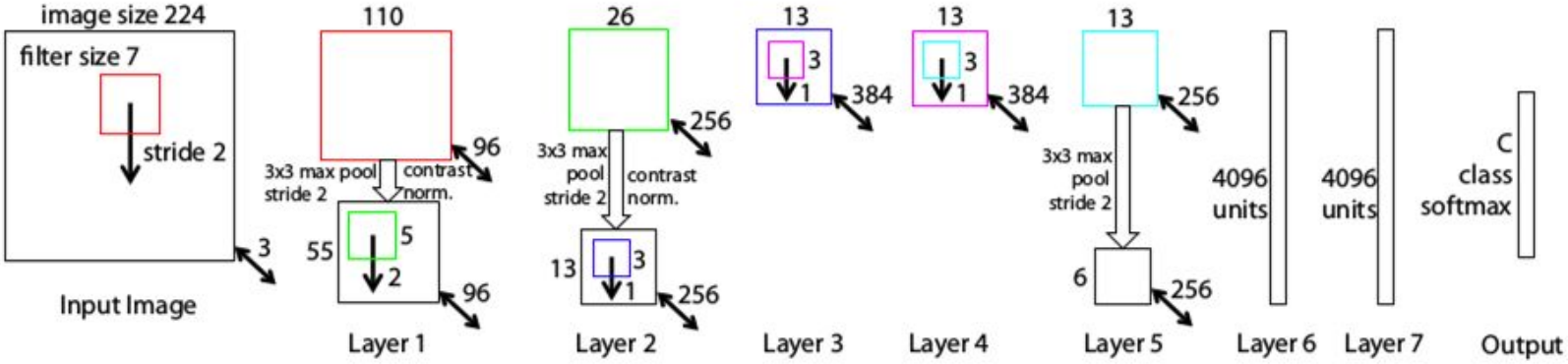
Girish Varma
IIIT Hyderabad

Alexnet



Dropouts at fully connected layer

ZFNet Matthew D Zeiler, Rob Fergus

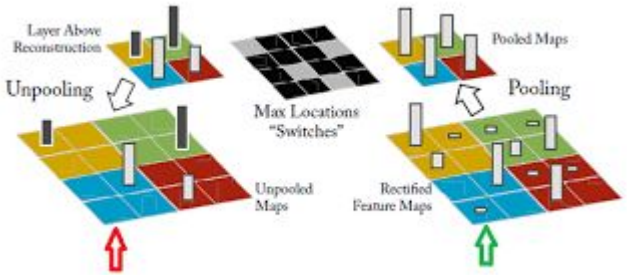


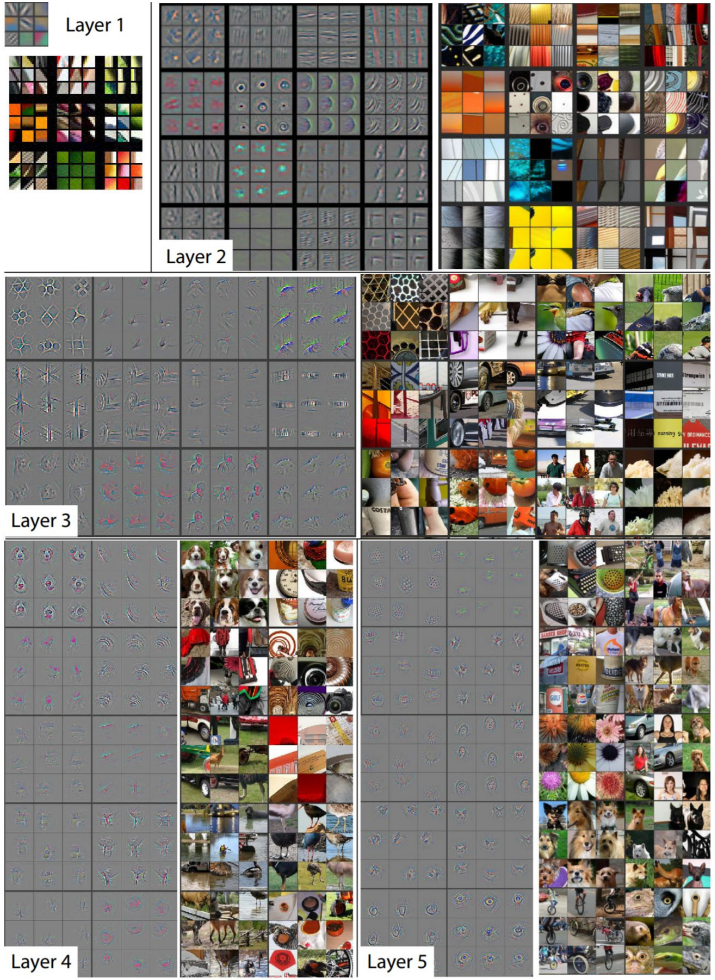
ZF Net Architecture

Winner of ILSVRC 2013 Classification Task.

ZFNet

Interpreting the feature activity in intermediate layers, by mapping these activities back to the input pixel space, by using deconv layers (unpooling + transpose filtering)





ZFNet : Input Jittering

1. Each RGB image was preprocessed by resizing the smallest dimension to 256,
2. cropping the center 256x256 region,
3. subtracting the per-pixel mean (across all images)
4. then using 10 different sub-crops of size 224x224 (corners + center with(out) horizontal flips).

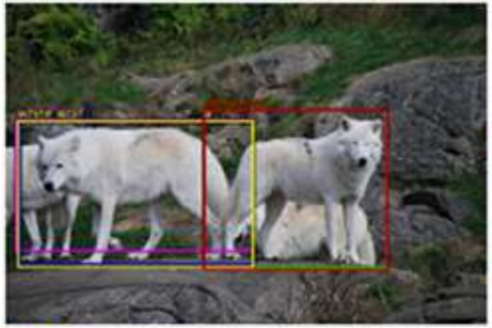
OverFeat

Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, Yann LeCun

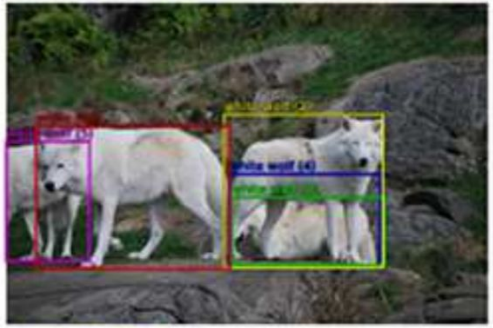
Layer	1	2	3	4	5	6	7	Output 8
Stage	conv + max	conv + max	conv	conv	conv + max	full	full	full
# channels	96	256	512	1024	1024	3072	4096	1000
Filter size	11x11	5x5	3x3	3x3	3x3	-	-	-
Conv. stride	4x4	1x1	1x1	1x1	1x1	-	-	-
Pooling size	2x2	2x2	-	-	2x2	-	-	-
Pooling stride	2x2	2x2	-	-	2x2	-	-	-
Zero-Padding size	-	-	1x1x1x1	1x1x1x1	1x1x1x1	-	-	-
Spatial input size	231x231	24x24	12x12	12x12	12x12	6x6	1x1	1x1

- A bottleneck in improving performance is size of dataset.
- Training to classify, locate and detect objects improves accuracy of all these tasks
- Run the Classifier at different crops and scales

Overfeat



Top 5:
white wolf
white wolf
timber wolf
timber wolf
Arctic fox



Groundtruth:
white wolf
white wolf (2)
white wolf (3)
white wolf (4)
white wolf (5)

Top predictions:
person (confidence 6.0)

Groundtruth:
drum
lamp
lamp (2)
guitar

GoogleNet

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich

Winner of ILSVRC 2014 Challenge

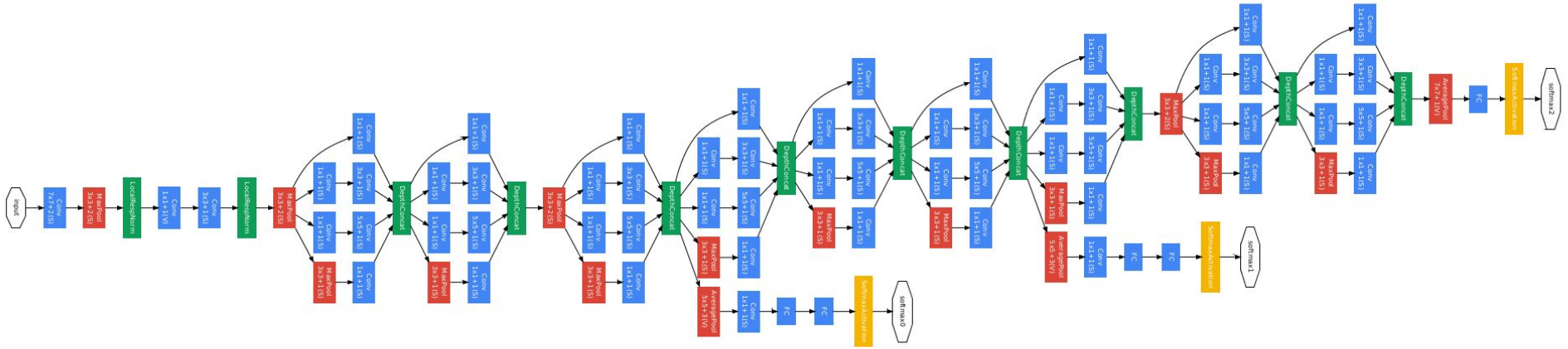
Deeper Convolutions

More efficient use of parameters

1. Reduced parameters in the fully connected layer.
2. Careful design of Convolution layers called Inception Modules.

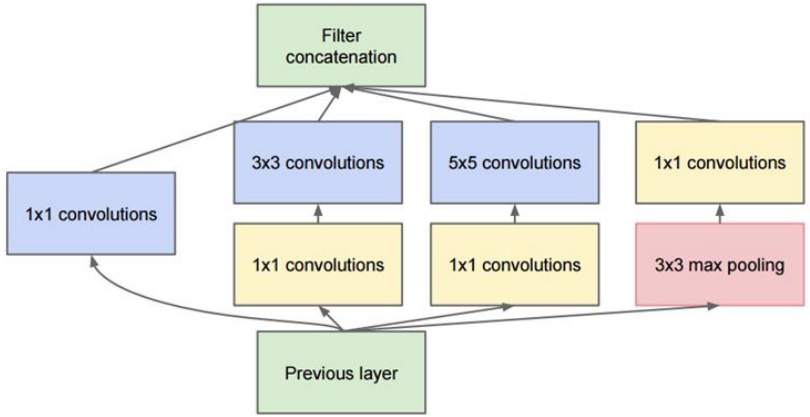
12x fewer parameters than Alexnet, with improved accuracy.

GoogleNet Going Deeper with Convolutions



GoogleNet : Inception Modules

1×1 convolutions are used to compute reductions before the expensive 3×3 and 5×5 convolutions.



Full Inception module

GoogleNet : Inception Modules

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

VGGNet

deeper-is-better philosophy

3x3 conv. kernels – very small

Stacked conv. layers have a large receptive field:

two 3x3 layers – 5x5 receptive field

three 3x3 layers – 7x7 receptive field

3 fully-connected (FC) layers

140 Milllion Parameters!!

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

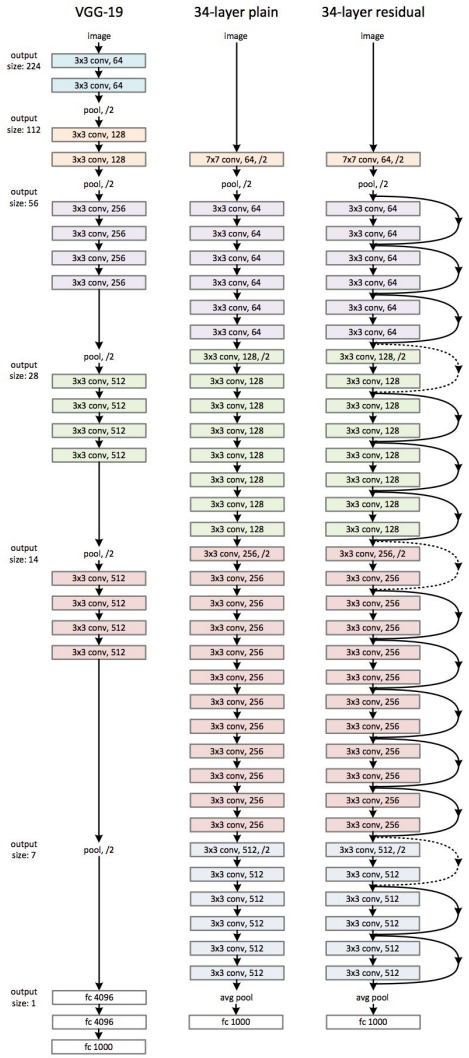
Table 2: **Number of parameters** (in millions).

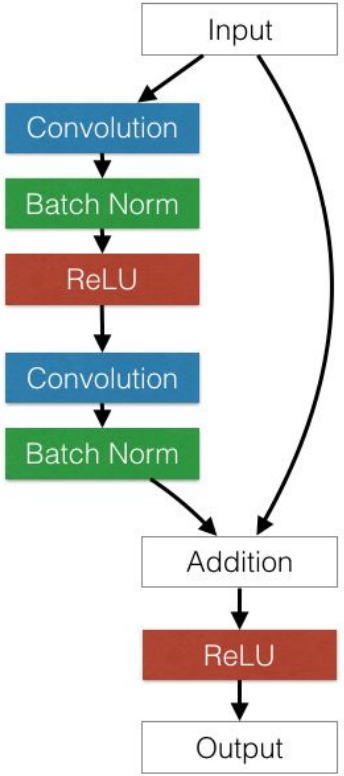
Network	A,A-LRN	B	C	D	E
Number of parameters	133	133	134	138	144

ResNet

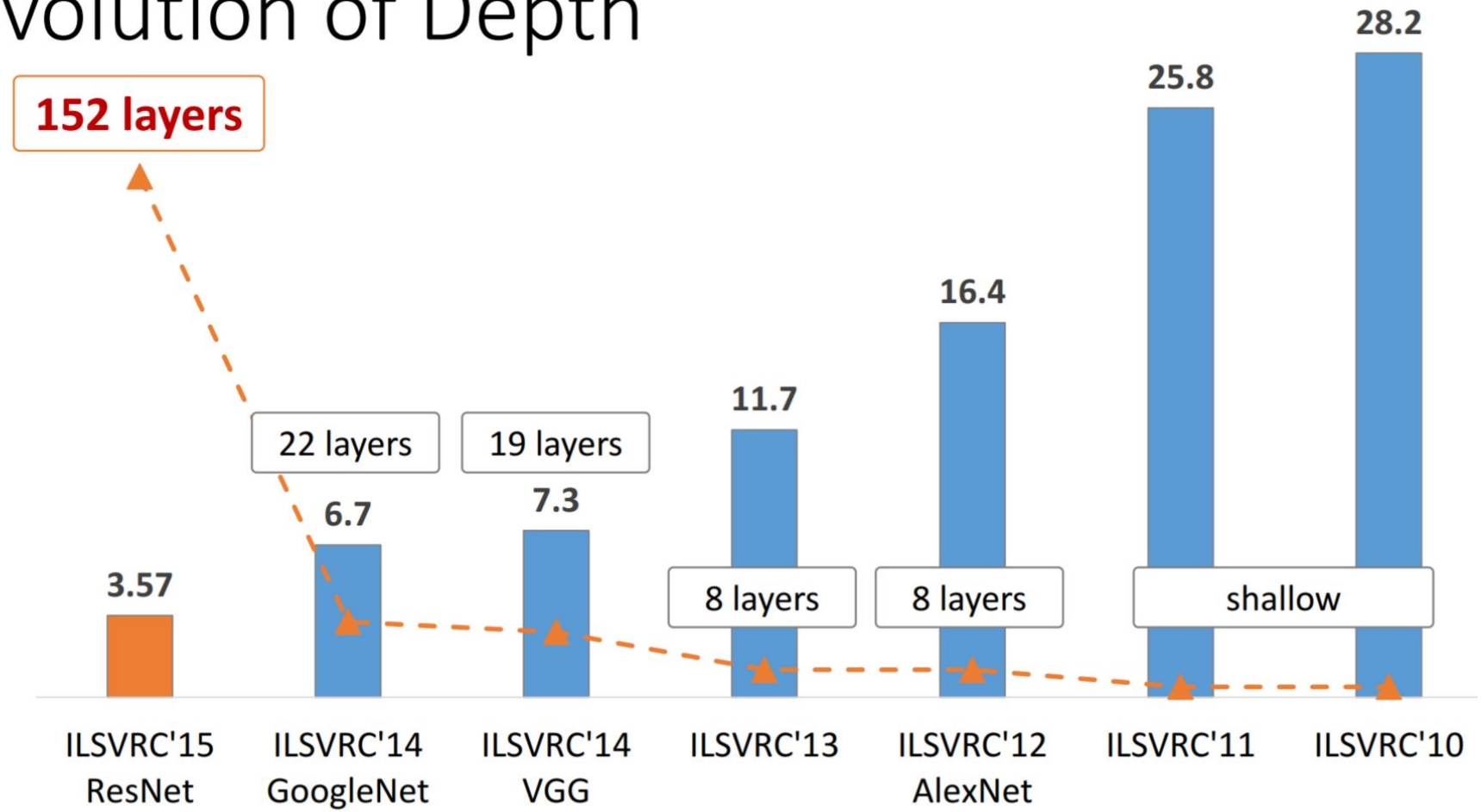
Deep Networks have a vanishing gradient problem.

Resnet overcomes it using Skip connections.

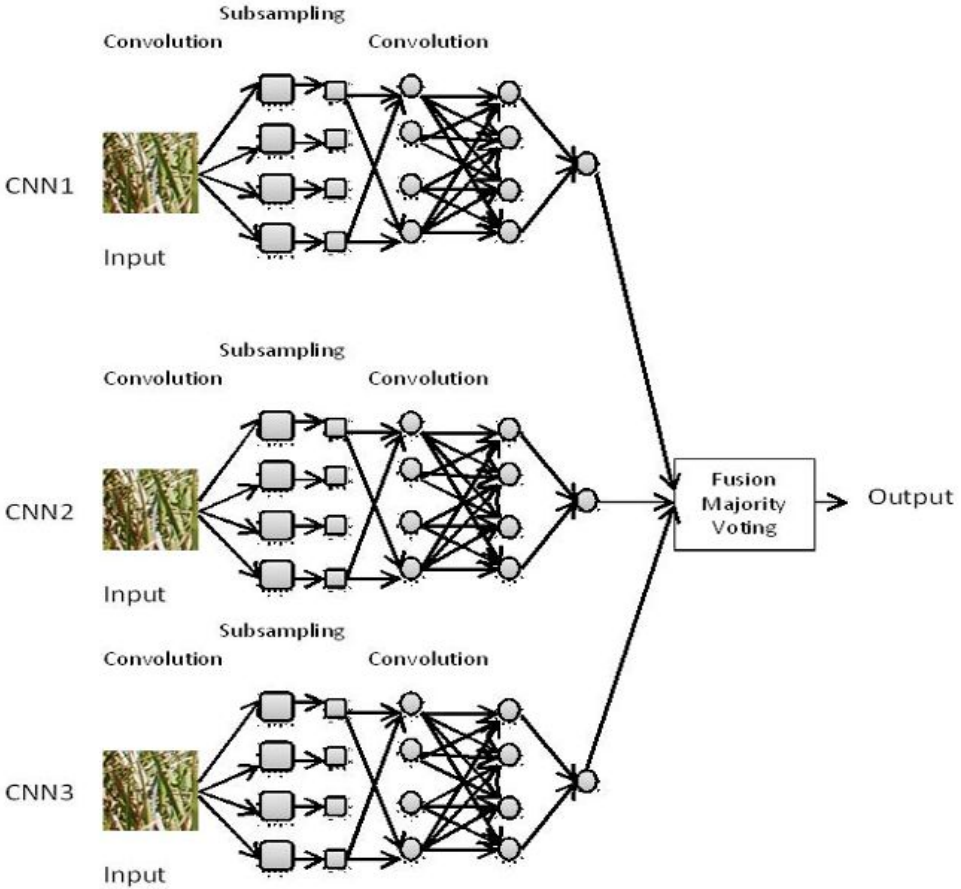




Revolution of Depth



Ensembles


Subsampling
Convolution
Convolution
CNN1
Input
Subsampling
Convolution
Convolution
CNN2
Input
Subsampling
Convolution
Convolution
CNN3
Input
Fusion
Majority
Voting
Output

Beyond Classification

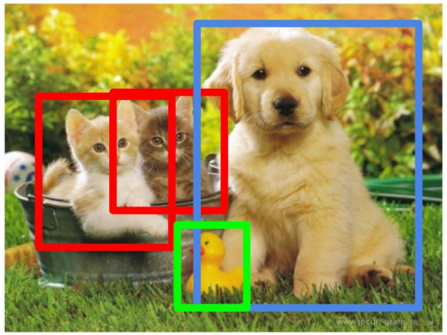
Classification

CAT

Classification + Localization

CAT

Object Detection



CAT, DOG, DUCK

Instance Segmentation

CAT, DOG, DUCK

Single object

Multiple objects

Classification + Localization: Task

Classification: C classes
Input: Image
Output: Class label
Evaluation metric: Accuracy

CAT

Localization:
Input: Image
Output: Box in the image (x, y, w, h)
Evaluation metric: Intersection over Union

(x, y, w, h)

Idea #1: Localization as Regression

Input: image

Only one object,
simpler than detection

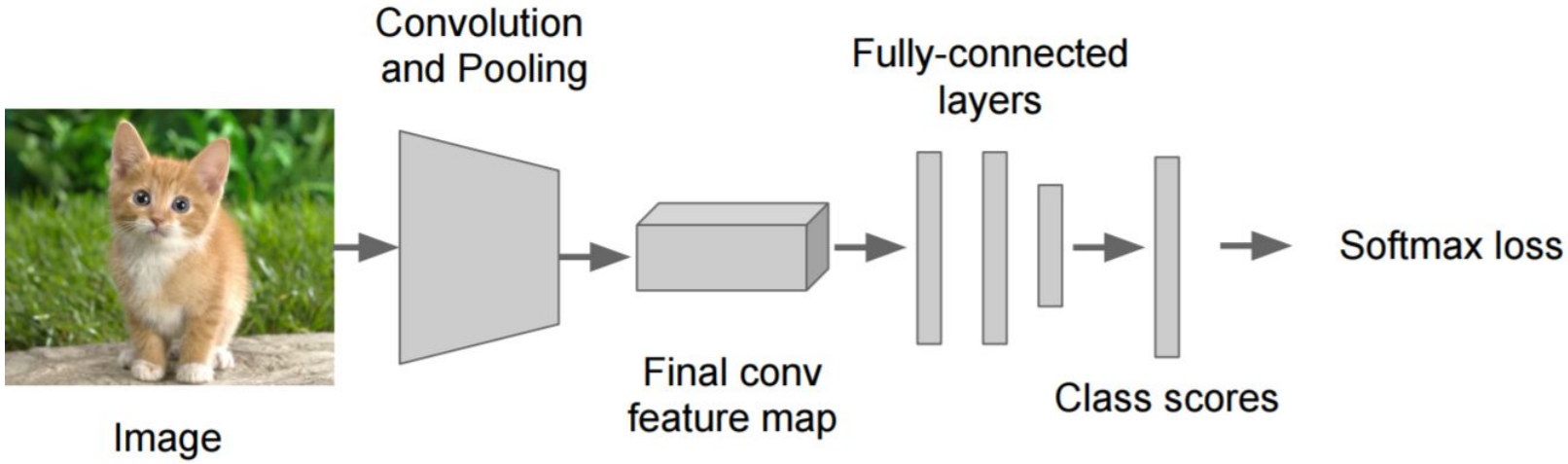
Neural Net

Output:
Box coordinates
(4 numbers)

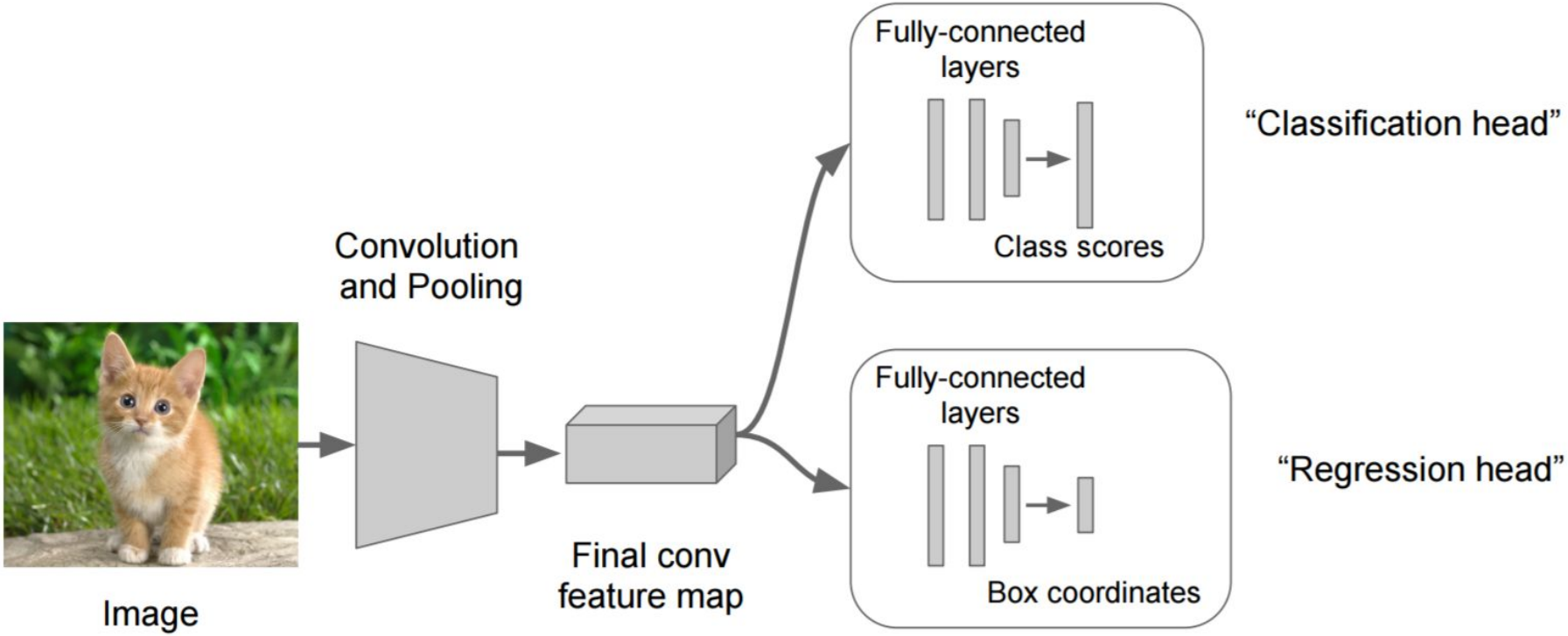
Correct output:
box coordinates
(4 numbers)

Loss:
L2 distance

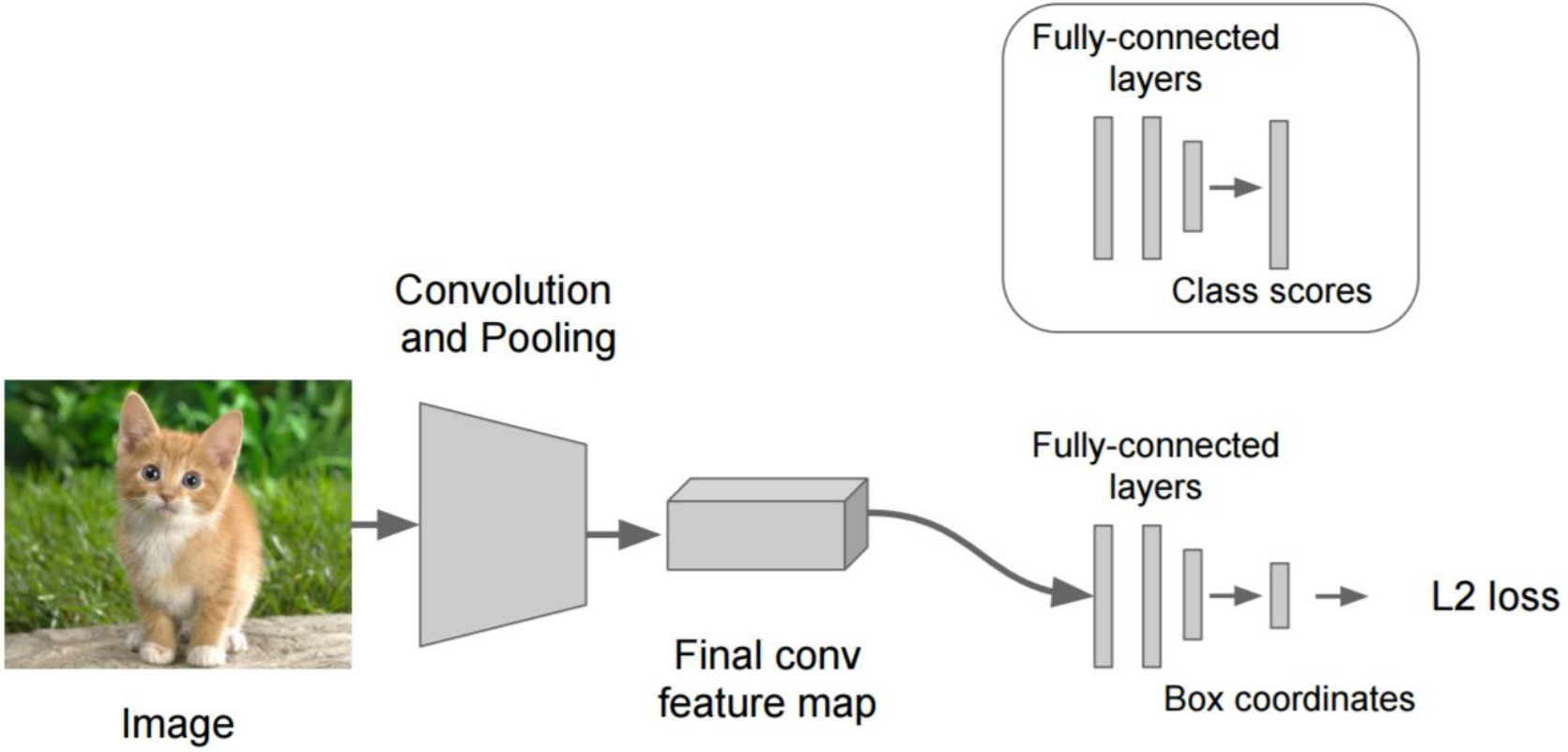
Step 1: Train (or download) a classification model (AlexNet, VGG, GoogLeNet)



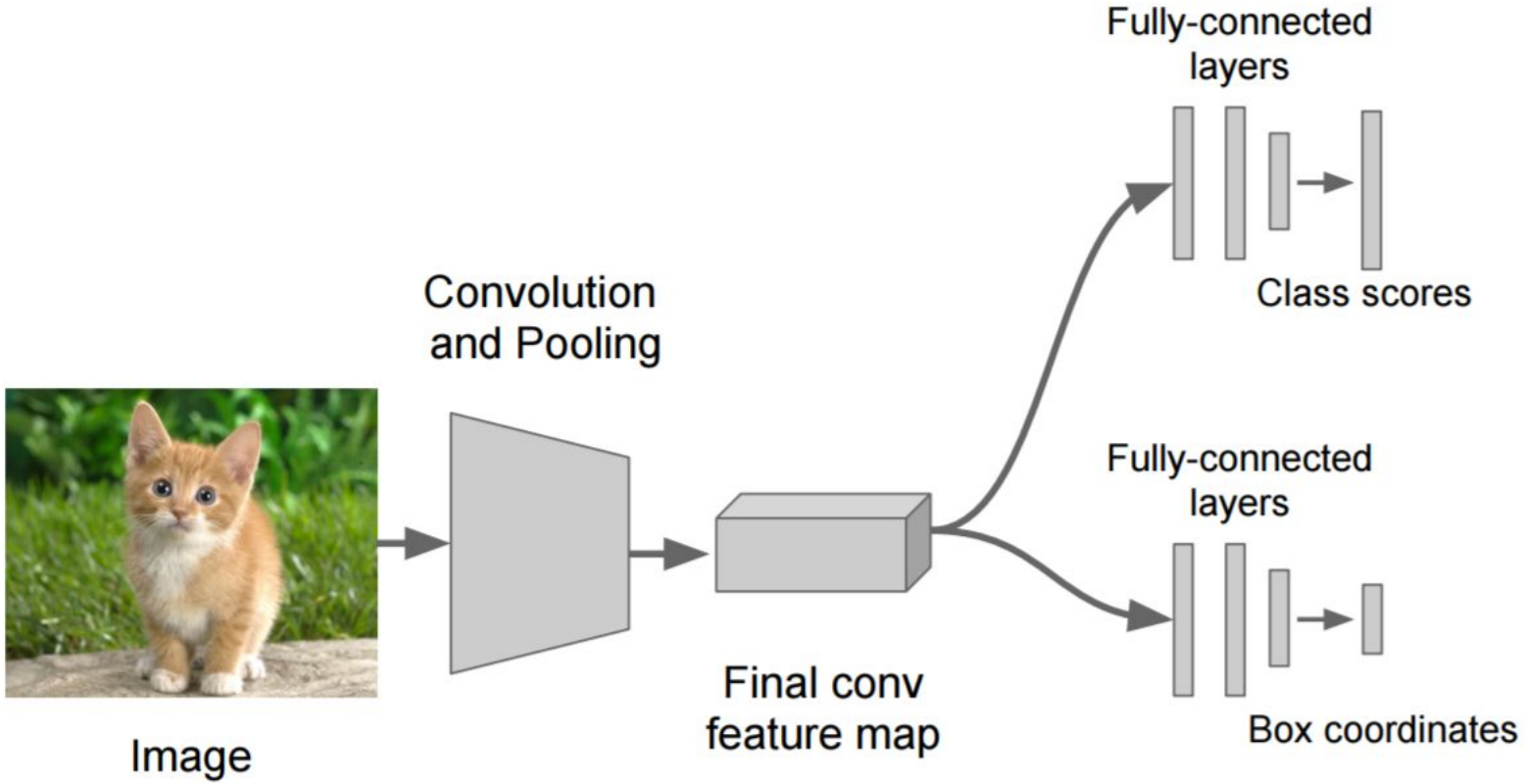
Step 2: Attach new fully-connected “regression head” to the network



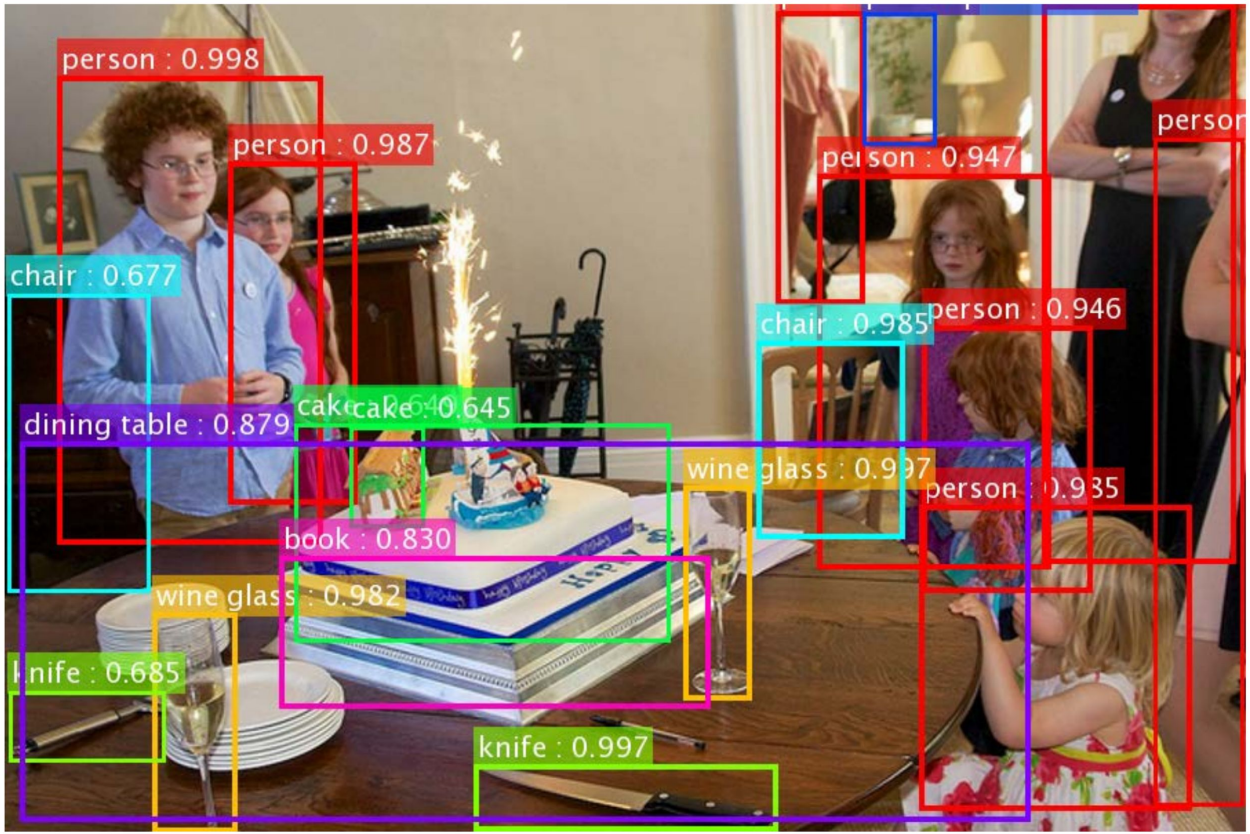
Step 3: Train the regression head only with SGD and L2 loss



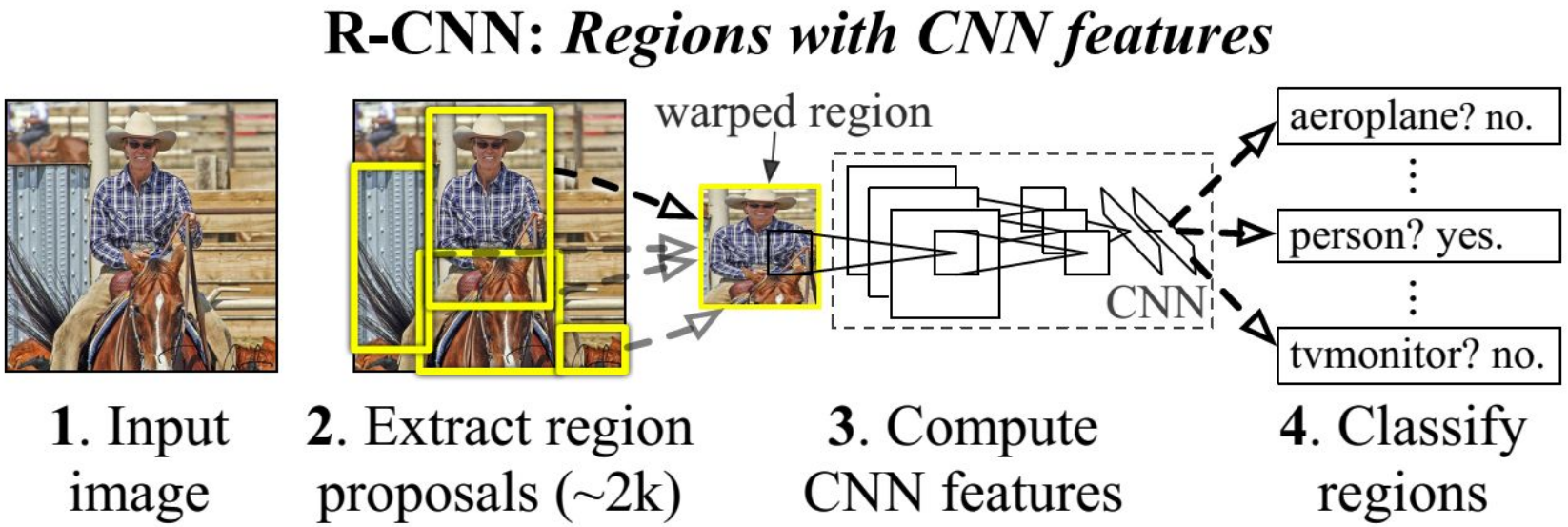
Step 4: At test time use both heads



Detection


person : 0.998
person : 0.987
person : 0.947
chair : 0.677
person : 0.946
chair : 0.985
dining table : 0.879
cake : 0.645
wine glass : 0.997
person : 0.935
book : 0.830
wine glass : 0.982
knife : 0.685
knife : 0.997

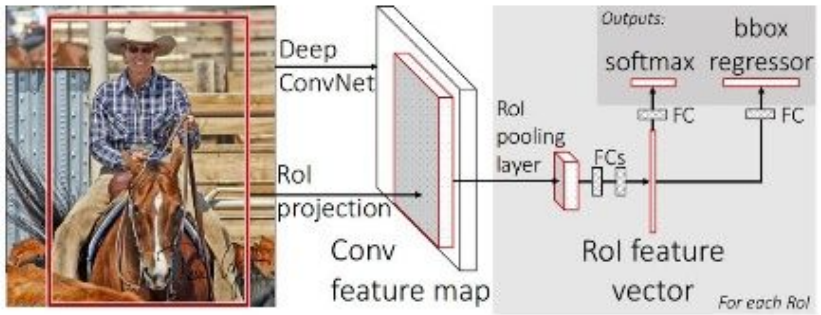
Detection


R-CNN: Regions with CNN features
warped region
CNN
aeroplane? no.
person? yes.
tvmonitor? no.
1. Input image
2. Extract region proposals (~2k)
3. Compute CNN features
4. Classify regions

Detection Faster

Fast R-CNN

R-CNN Problem #1: Slow at test-time: need to run full forward pass of CNN for each region proposal

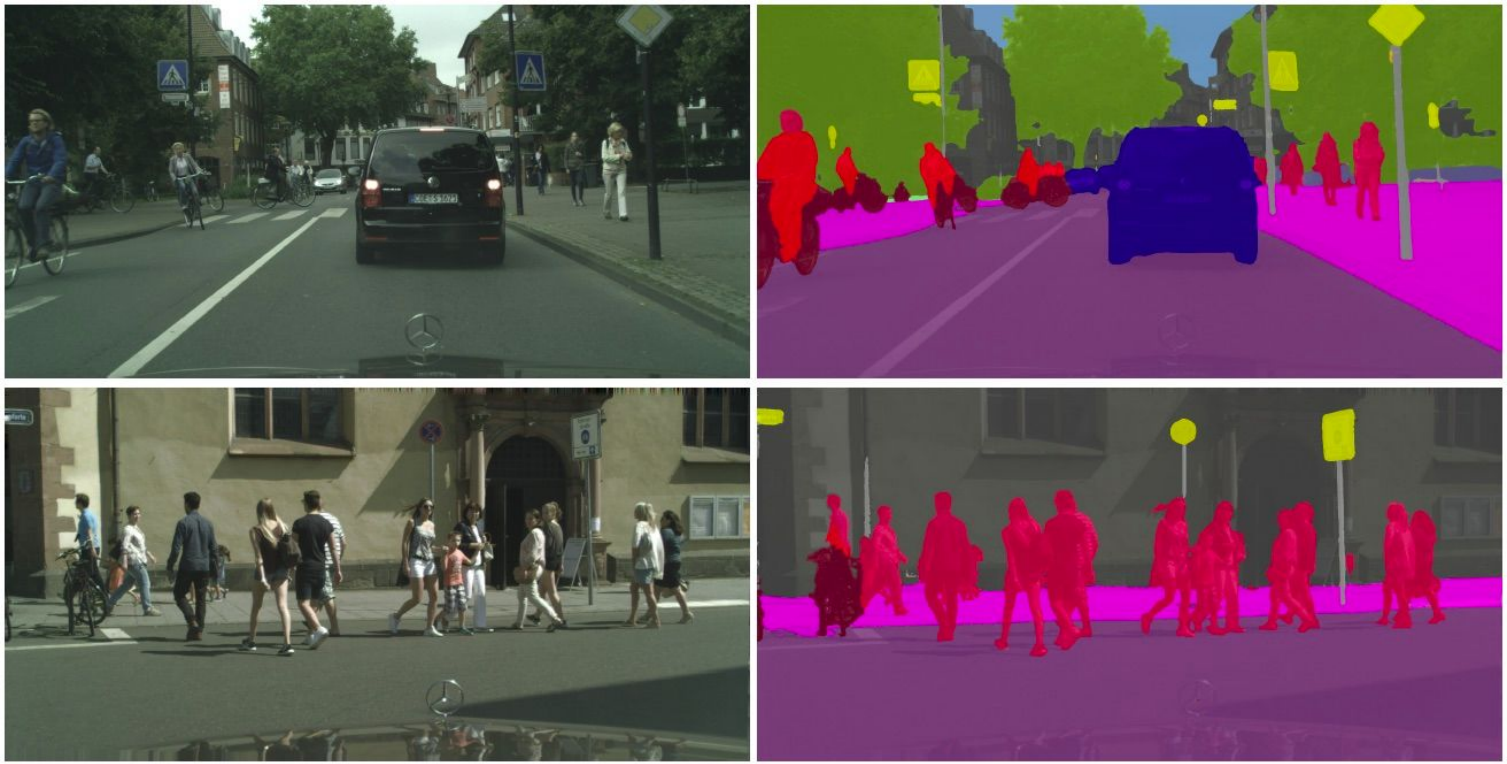


Solution: Share computation of convolutional layers between region proposals for an image

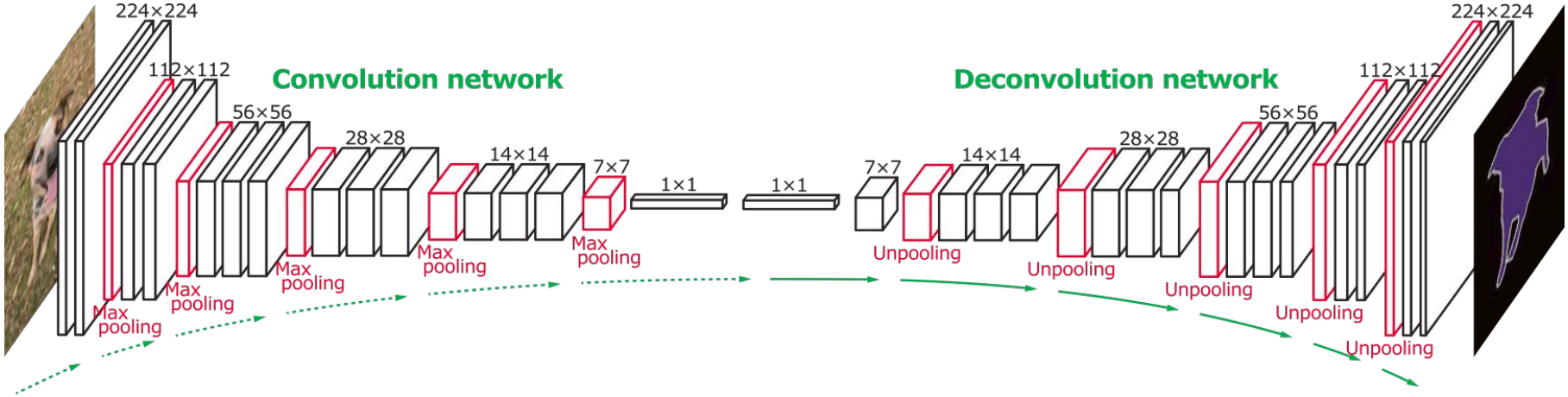
Girshick Fast R-CNN. ICCV 2015

Also makes use of multi task loss function.

Semantic Segmentation

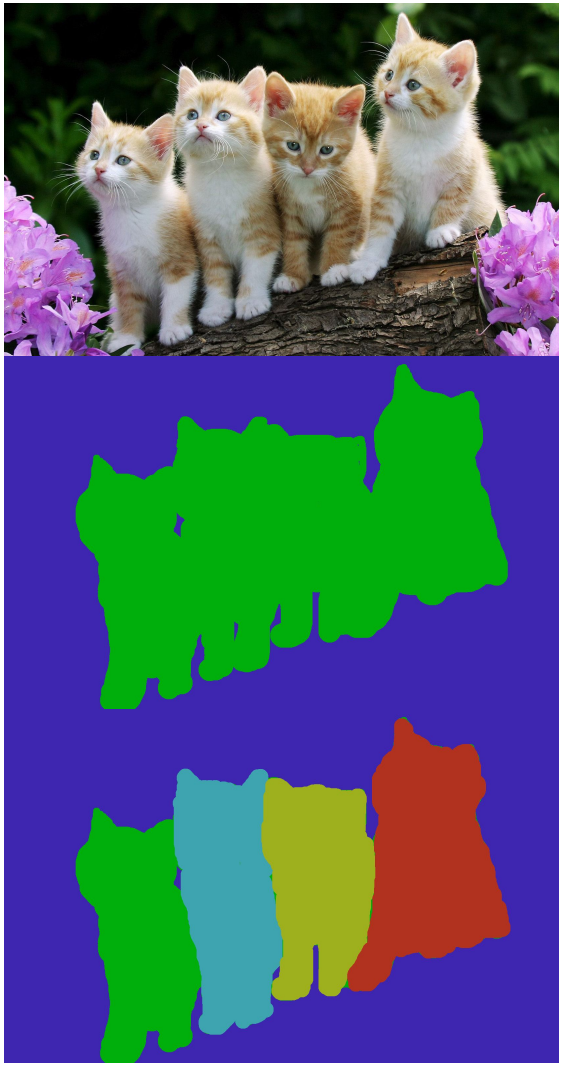


Semantic Segmentation



Fully Connected as a convolution : Fully Conv Networks

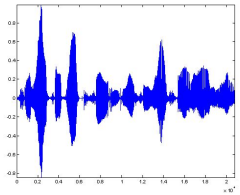
Instance Segmentation



Sequence to Sequence Problems

Convert $x_1,x_2,...,x_s$ to $y_1,y_2,...,y_t$

Speech to Text: speech signal x(i) to text y(i)



→ Hello

Language Translation: text in language 1 x(i) to text in language 2 y(i)

बुश की 77 करोड़ डॉलर की मदद की पेशकश

अमरीकी राष्ट्रपति बुश ने दुनियाभर में खाद्य पदार्थों की बढ़ती क़ीमतों के असर को कम करने के लिए 77 करोड़ डॉलर की खाद्य सहायता की पेशकश की है.
+ खाद्यान्न संकट टास्क फ़ोर्स का गठन

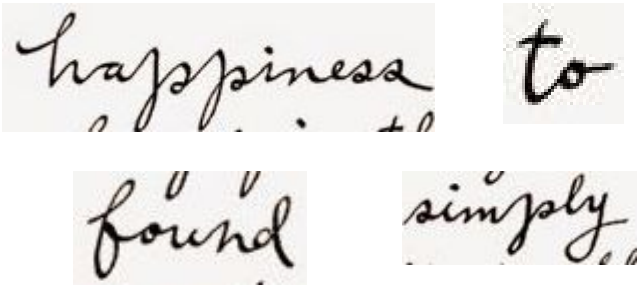
Bush offered the help of a $ 77 million

U.S. President George Bush has in the world of food products to reduce the impact of rising prices for 77 million dollars in food aid offered.

Sequence to Sequence Problems

Convert $x_1,x_2,...,x_s$ to $y_1,y_2,...,y_t$

OCR : Convert these images to unicode strings

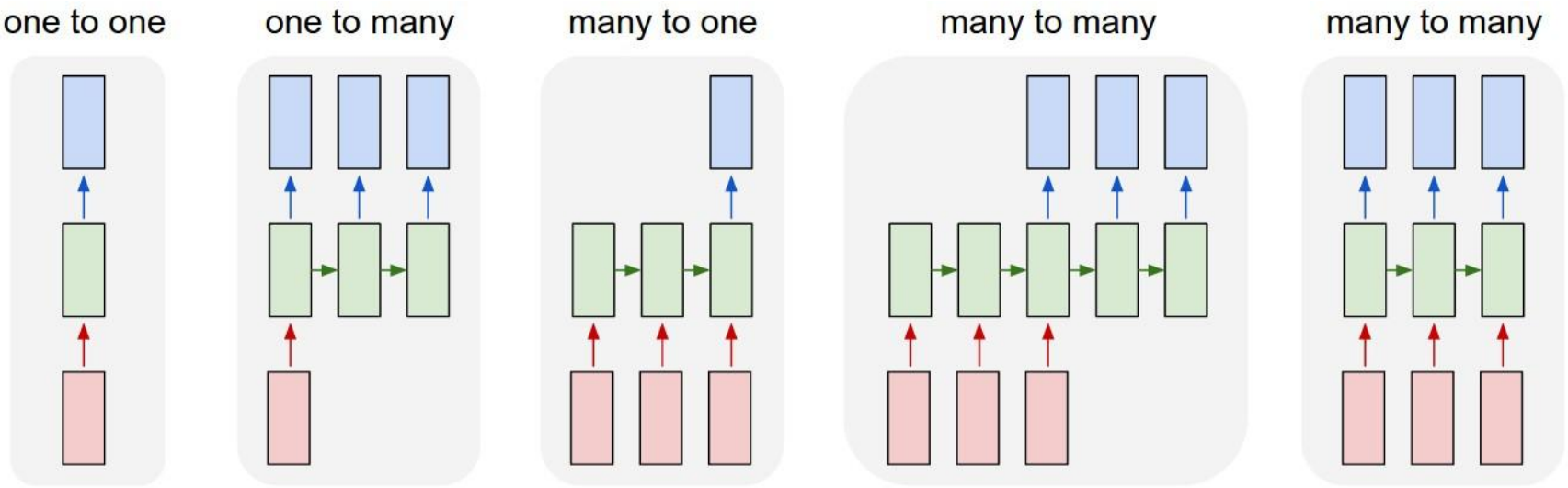


Time series data : Given a sequence predict the next element

- Predict stock prices
- Weather prediction

Sequence to Sequence Problems

Convert $x_1, x_2, ..., x_s$ to $y_1, y_2, ..., y_t$

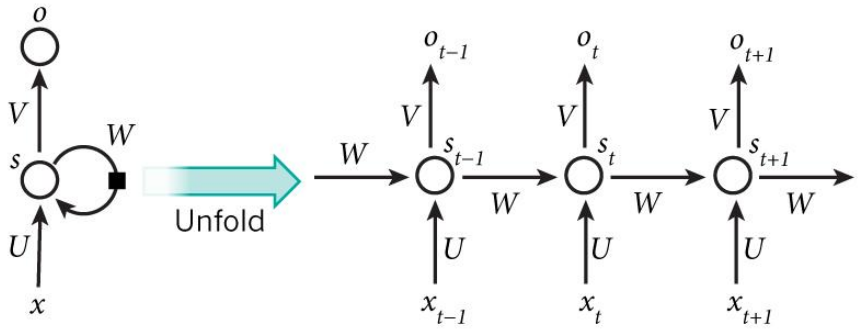


Recurrent Neural Networks

Convert $x_1,x_2,...,x_s$ to $o_1,o_2,...,o_t$

$s_t = \tanh(Ux_t + Ws_{t-1} + b)$

$O_t = Vs_t$



Scenetxt

CNN + RNN

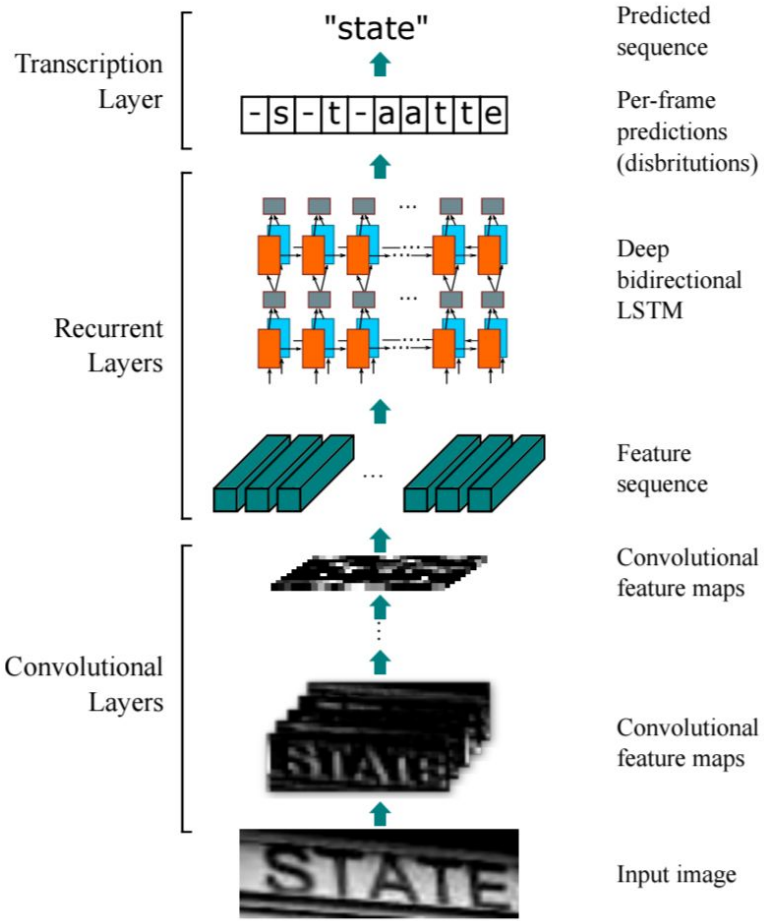
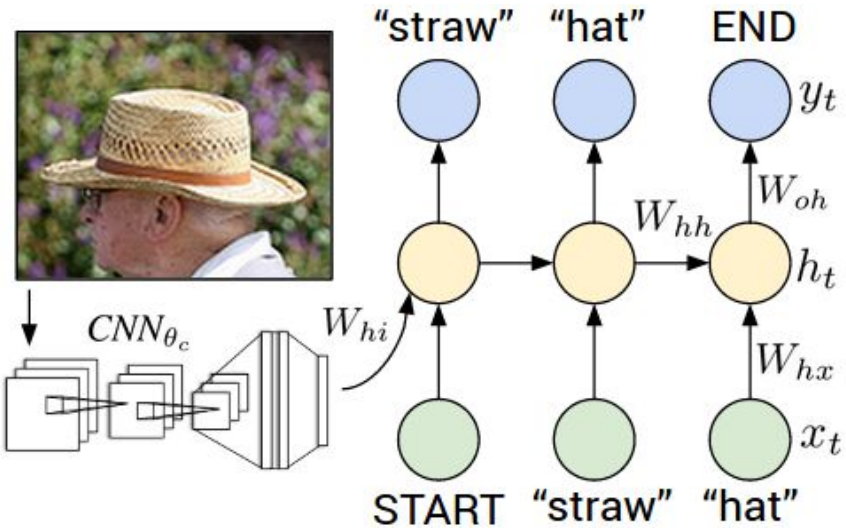
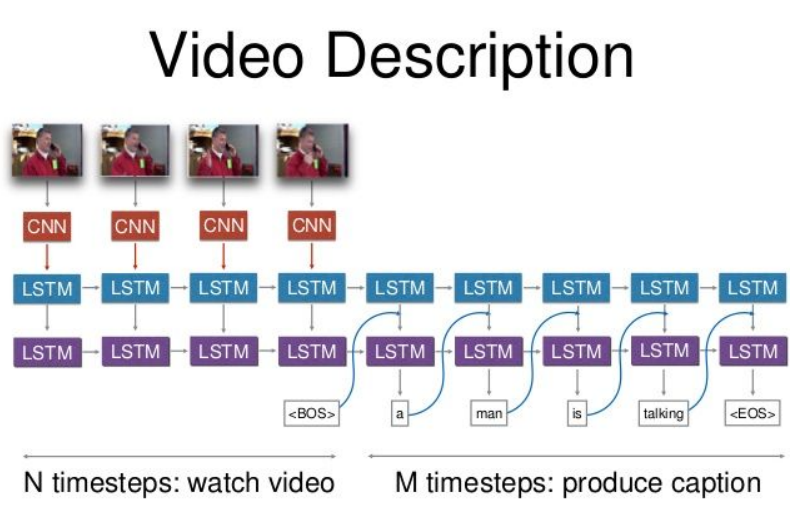


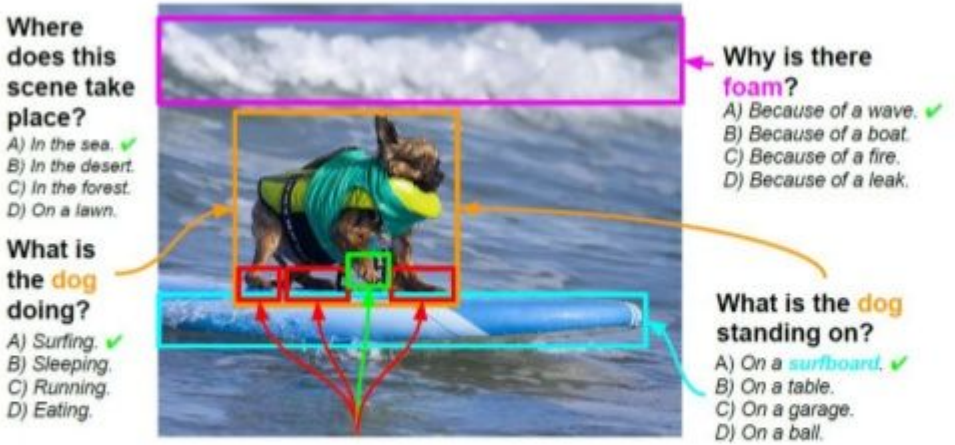
Image & Video Captioning





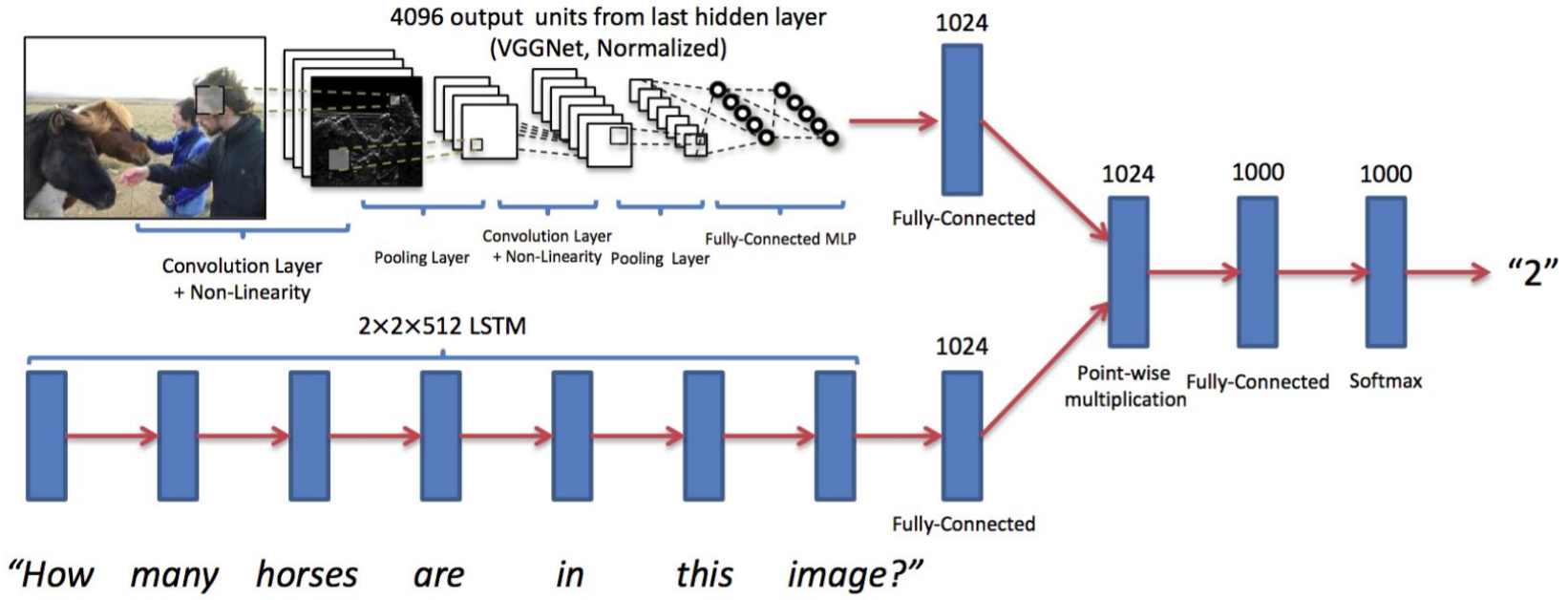
Venugopalan et al., "Sequence to Sequence -- Video to Text," 2015.
http://arxiv.org/abs/1505.00487

Visual Question Answering



(Slides and Screencast by Issey Masuda): Zhu, Yuke, Oliver Groth, Michael Bernstein, and Li Fei-Fei."Visual7W: Grounded Question Answering in Images." CVPR 2016.

Visual Question Answering



Reference

http://cs231n.stanford.edu/slides/2017 (Must Read) Lectures 8