Deep Neural Networks

Machine Learning and Pattern Recognition

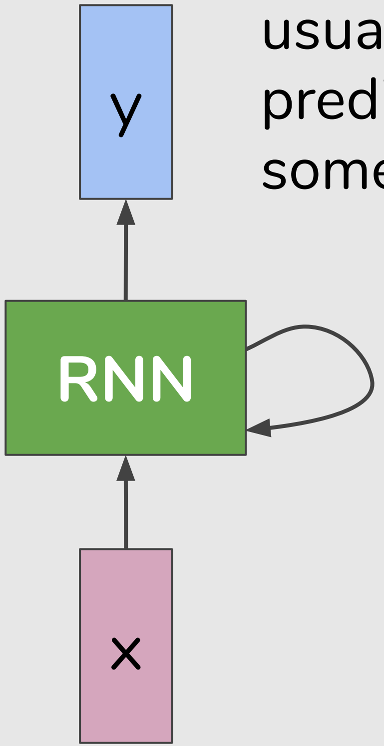
(Largely based on slides from Luis Serrano & Fei-Fei Li & Andrej Karpathy & Justin Johnson & Serena Yeung)

Prof. Sandra Avila

Institute of Computing (IC/Unicamp)

MC886/MO444, October 30, 2018

Recurrent Neural Network

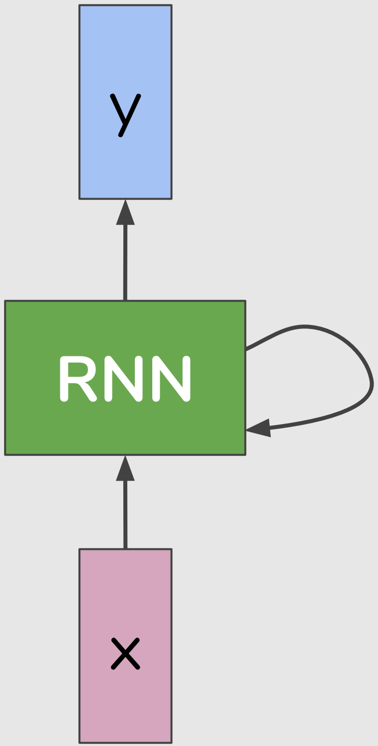


usually want to predict a vector at some time steps

Recurrent Neural Network

We can process a sequence of vectors **x** by applying a **recurrence formula** at every time step:

$$h_t = f_W(h_{t-1}, x_t)$$



Recurrent Neural Network

We can process a sequence of vectors **x** by applying a **recurrence formula** at every time step:

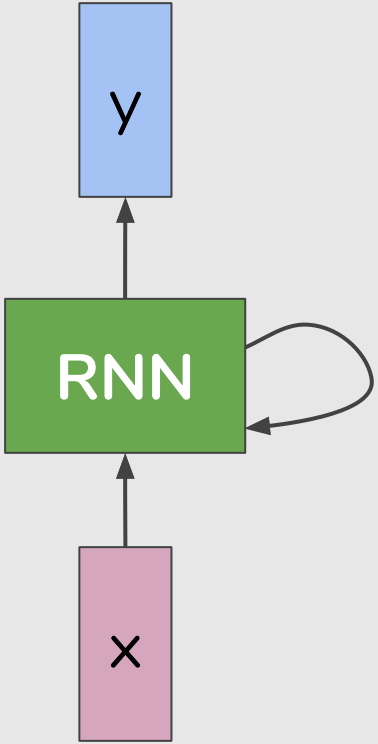
$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

new state

old state

input vector at
some time step

some function
with parameters W



Recurrent Neural Network

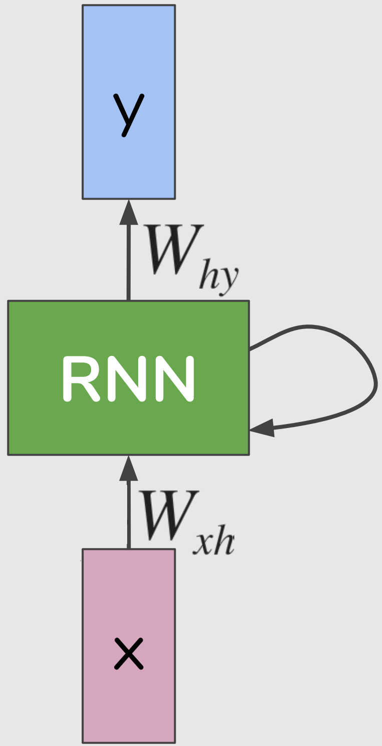
The state consists of a single "hidden" vector **h**:

$$h_t = f_W(h_{t-1}, x_t)$$

$$\Downarrow$$

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$



Recurrent Neural Networks: Process Sequences

one to one

Vanilla Neural Networks

one to many

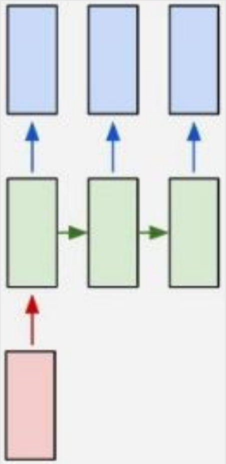
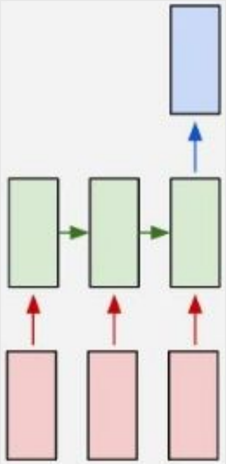


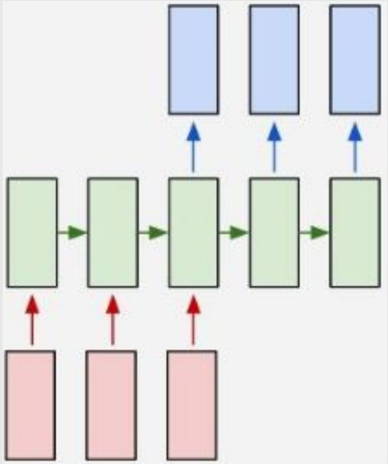
Image Captioning
image ⇒ seq. words

many to one



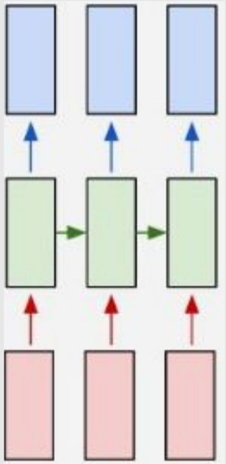
Sentiment Classification
seq. words ⇒ sentiment

many to many



Machine Translation
seq. words ⇒ seq. of words

many to many



Video classification on frame level

Training: “Maior dúvida da aula” 27/october/2017

https://github.com/llSourcell/recurrent_neural_network

GoogLeNet, Inception Module

Não entendi muito bem sobre as inception layers na GoogLeNet. Entendi a ideia de fazer a mesma coisa de um filtro grande com vários filtros menores. Com vários filtros menores temos menos parâmetros que um filtro grande?

Quando fazemos inception e concatenados os resultados, podemos comparar isso à criação de vetor de características? Porque estamos retirando tipos diferentes de informações de uma mesma camada de input e juntando elas pra formar um output.

Acho que não consegui entender muito bem o inception module da arquitetura GoogLeNet. Para que ele serve exatamente? Obrigada.

no modelo de inception v4, usa a paralelizacao para obter menos parametros, entao esso quer dizer que enquanto menos parametros e mais profundo da melhores resultados?

Não entendi exatamente que fator possibilitou a remoção das camadas fully connected na GoogleLeNet. Pelo que eu entendi, as redes mais modernas voltaram com a camada fully connected. Então quando usá-la ou não usá-la?

Números de parâmetros

Em relação a arquiterua proposta na rede GoogLeNet, não ficou muito claro para mim as camadas internas, principalmente na parte em que aplicar vários filtros menores, equilave a aplicar um filtro maior (embora o resultado não seja o mesmo).

Não ficou claro para mim qual a vantagem de se utilizar, por exemplo, 3 pequenos filtros 3x3 ao invés de um 7x7. Na aula você comentou que é para evitar diminuir drasticamente a imagem, mas qual a desvantagem disso?

Eu nao entendi aquelas contas dos filtros que reduziam o numero de parametros

ResNet Filtro 1x1

Achei um pouco confuso as dimensões do filtro 1x1. Achei confuso a parte da convolução de tal filtro.

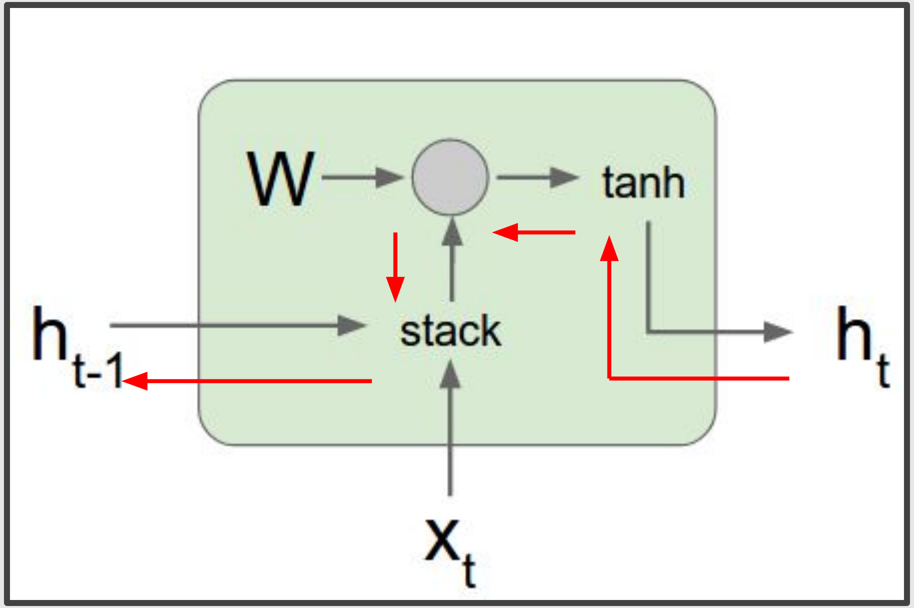
Training: “Maior dúvida da aula” 27/october/2017

```
iter 0, loss: 107.601633
----
 'ōqIE:ō:3(é
O Q.L"cÉhíL'uàfMO)êoâz.àãâéláç-)D(iéêdàF(lLFLrRcFA0nC(Pô(á#HM5éI?#ázHrtGTRF)5wlGaúa2éj?pd7,u
xp5LQ"r24F7é1efL"Cabvêúhy Ldã 7àã2àObmxv?qnAodí'P)mTg4(u4F7ú13ómrQnmeFNbãoúvâ3i?sxsuRãjáécó.-
záy
---
```

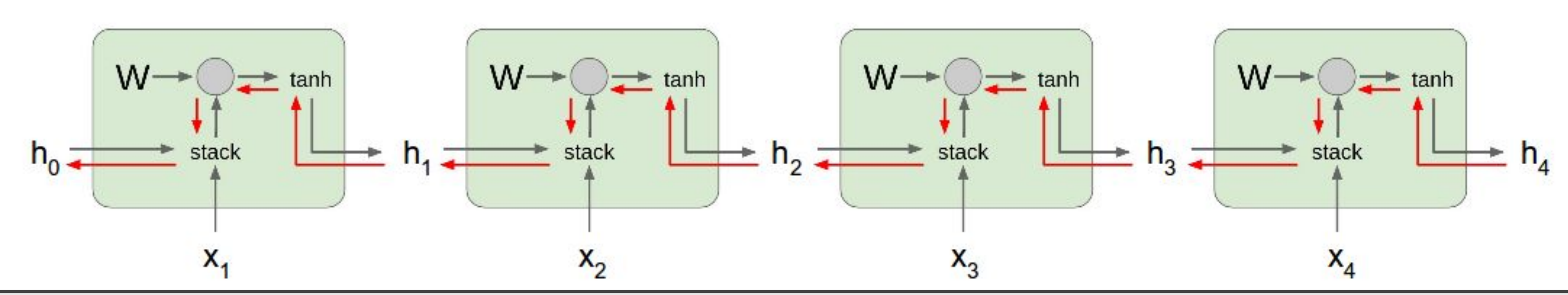
```
iter 46000, loss: 23.238596
----
 és GoogLeNet. E a rede aprede?

O Daras dúvrvilg. ( ende no pré-tro "rar outlara destidas? Com uttres dessar algo us filtros
parte novados aplicar au mula.

e nar
```

```
iter 204000, loss: 10.733449
----
 to, ina utir alpal asvelum motrio tarada mexexenterna mai reviso de enter meiss grandas

##### ResNet Filtro 1x1? Alheing?

Não entendi exatamente que fia, confenhalo deset desecta..

##### Como as
```

Vanilla RNN: Gradient Flow



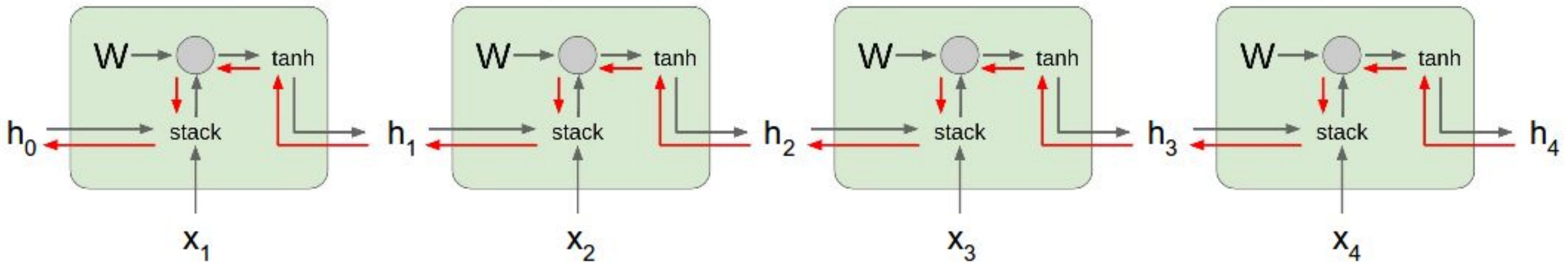
Vanilla RNN: Gradient Flow



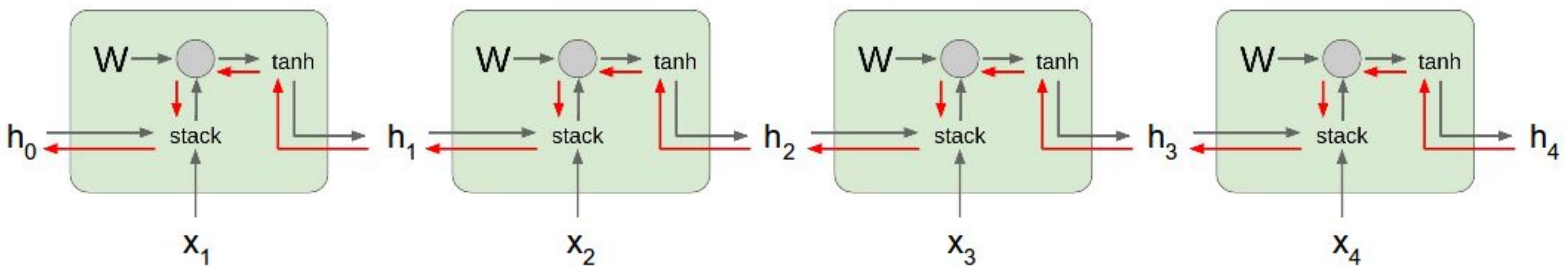
Largest singular value > 1:
Exploding gradients

Largest singular value < 1:
Vanishing gradients

Vanilla RNN: Gradient Flow



Vanilla RNN: Gradient Flow



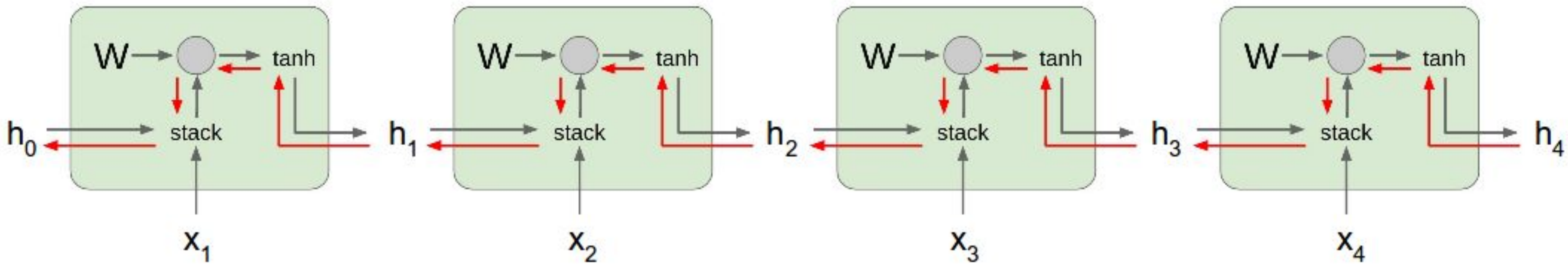
Largest singular value > 1:
Exploding gradients

Largest singular value < 1:
Vanishing gradients

Gradient clipping:
Scale gradient if its norm is too big.

```
grad_norm = np.sum(grad * grad)
if grad_norm > threshold:
  grad *= (threshold / grad_norm)
```

Vanilla RNN: Gradient Flow



Largest singular value > 1:
Exploding gradients

Largest singular value < 1:
Vanishing gradients

Change RNN architecture

Long Short Term Memory (LSTM)

Vanilla RNN

$$h_t = \tanh\left(W\begin{pmatrix}h_{t-1}\\x_t\end{pmatrix}\right)$$

LSTM

$$\begin{pmatrix}i\\f\\o\\g\end{pmatrix}=\begin{pmatrix}\sigma\\\sigma\\\sigma\\\tanh\end{pmatrix}W\begin{pmatrix}h_{t-1}\\x_t\end{pmatrix}$$

$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$

Hochreiter and Schmidhuber, “Long Short Term Memory”, Neural Computation, 1997

Long Short Term Memory (LSTM)

LSTM

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$

i : input gate, whether to write to cell

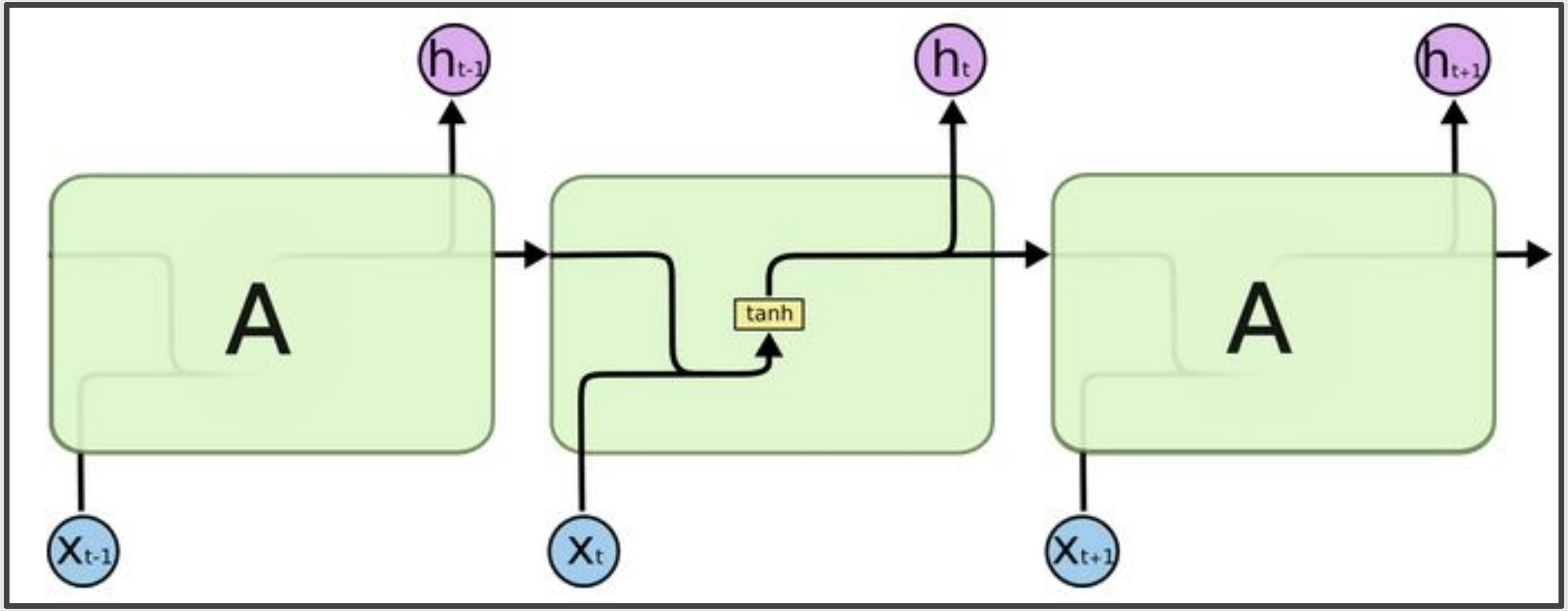
f : forget gate, whether to erase cell

o : output gate, how much to reveal cell

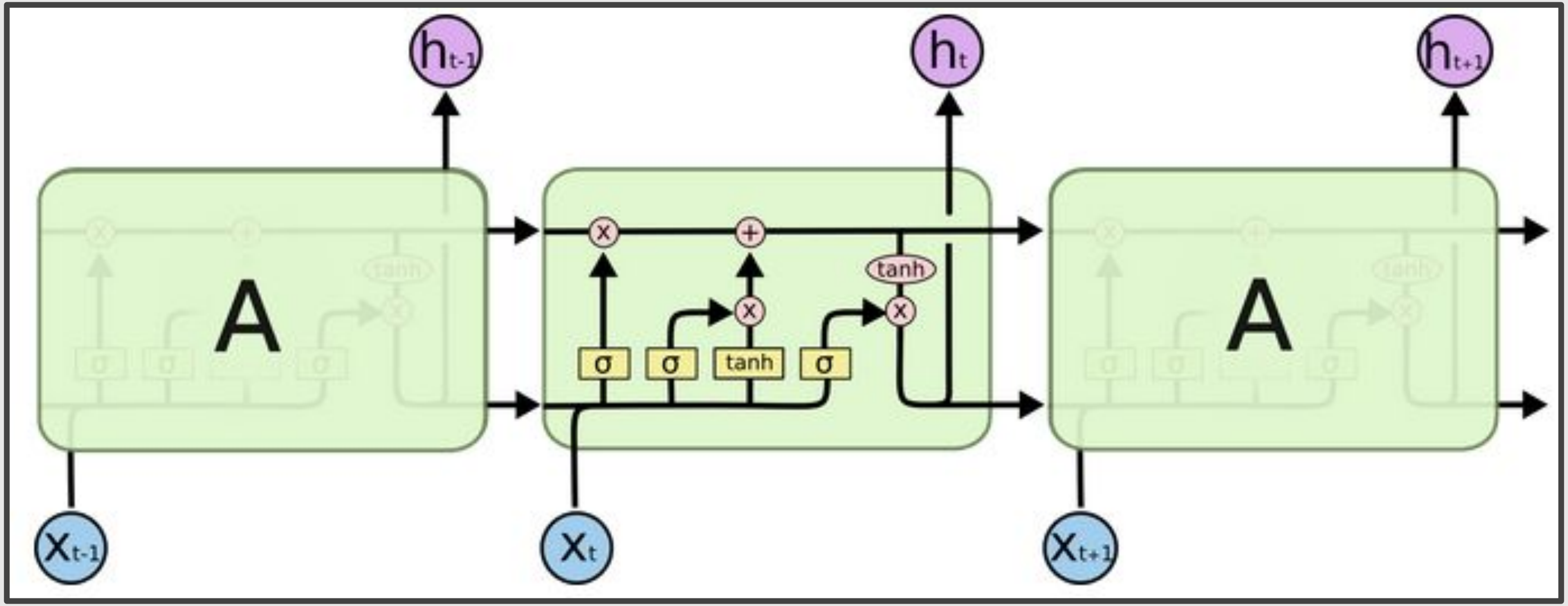
g : gate gate, how much to write to cell

Hochreiter and Schmidhuber, “Long Short Term Memory”, Neural Computation, 1997

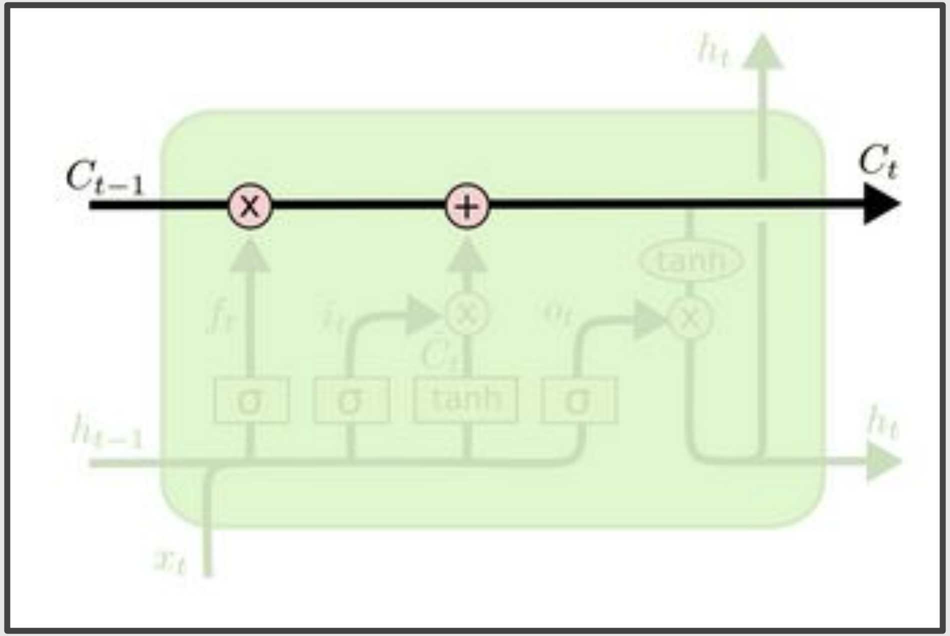
Long Short Term Memory (LSTM)



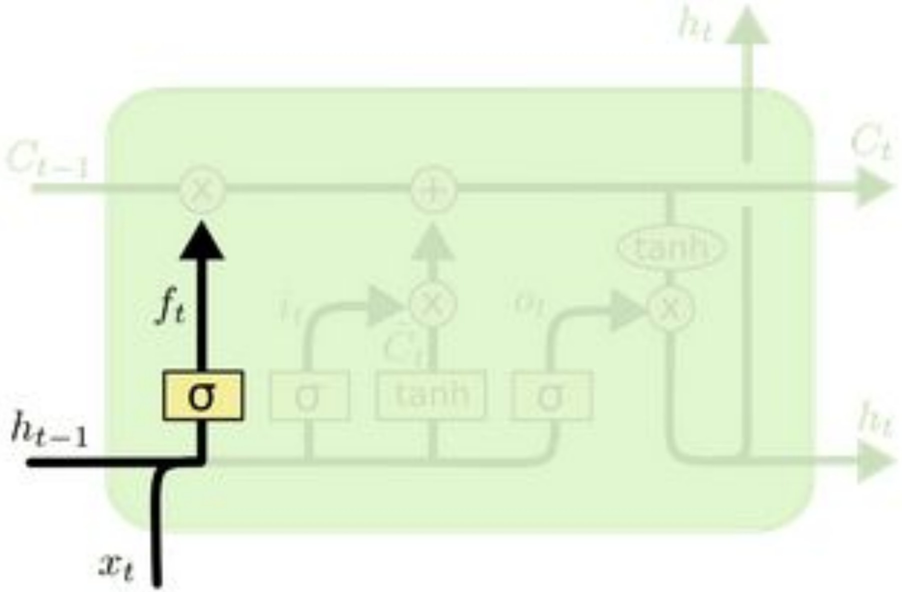
Long Short Term Memory (LSTM)



Long Short Term Memory (LSTM)



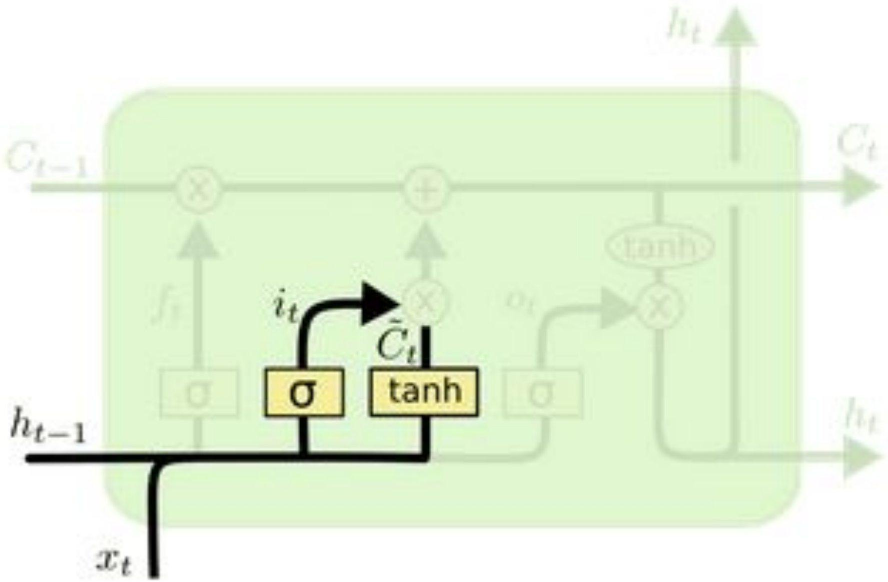
Long Short Term Memory (LSTM)



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

"forget gate layer"

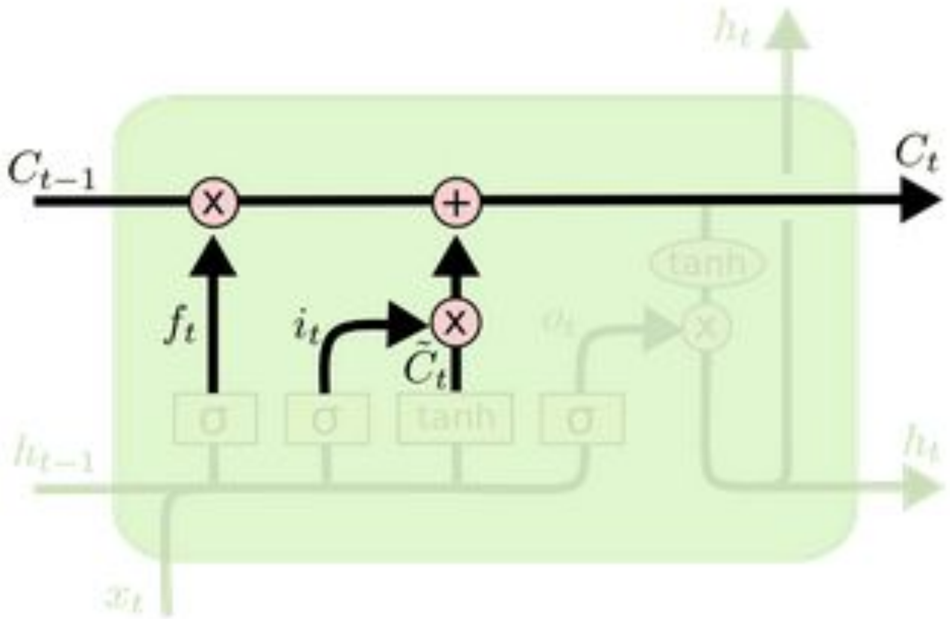
Long Short Term Memory (LSTM)



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

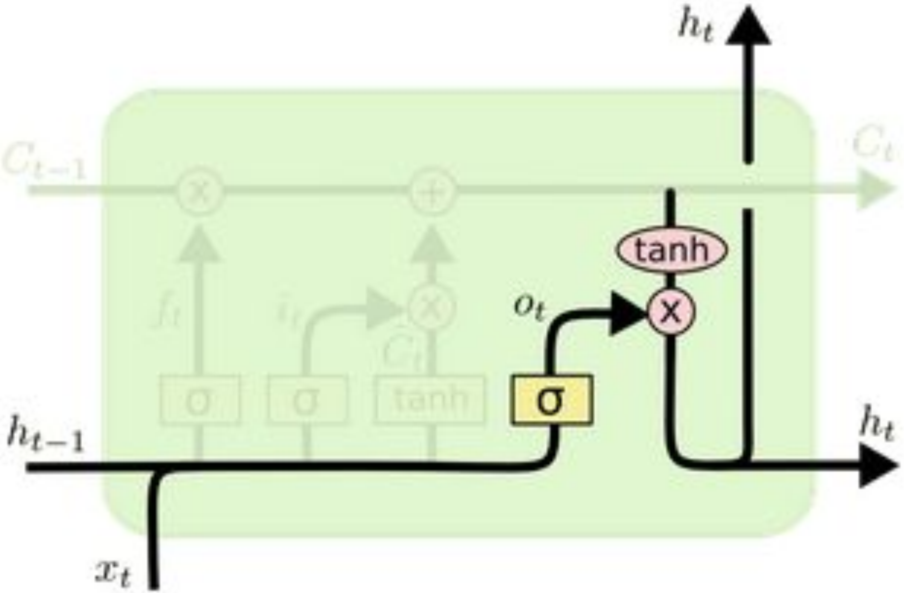
"input gate layer" decides which values we'll update

Long Short Term Memory (LSTM)



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

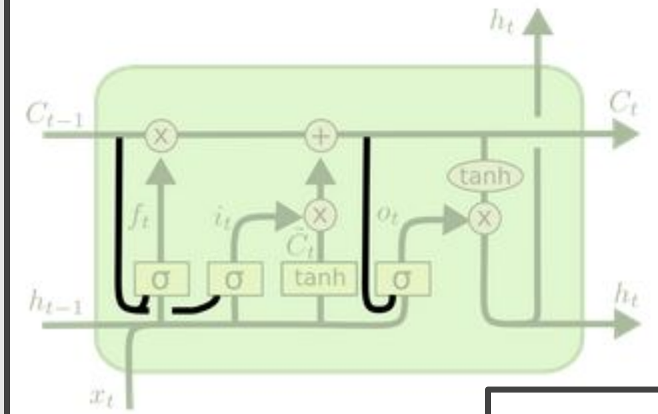
Long Short Term Memory (LSTM)



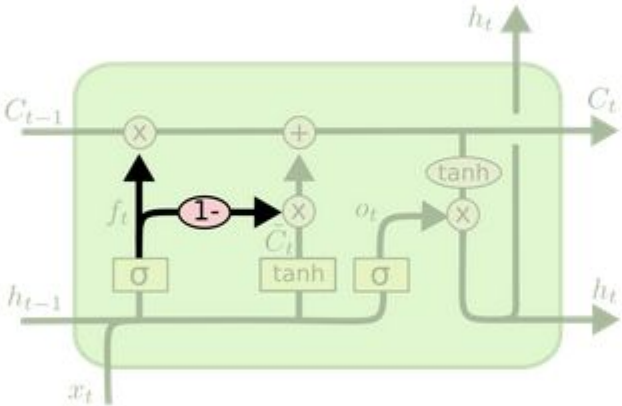
$$o_t = \sigma \left(W_o \ [h_{t-1}, x_t] \ + \ b_o \right)$$

$$h_t = o_t * \tanh \left(C_t \right)$$

LSTM Variations



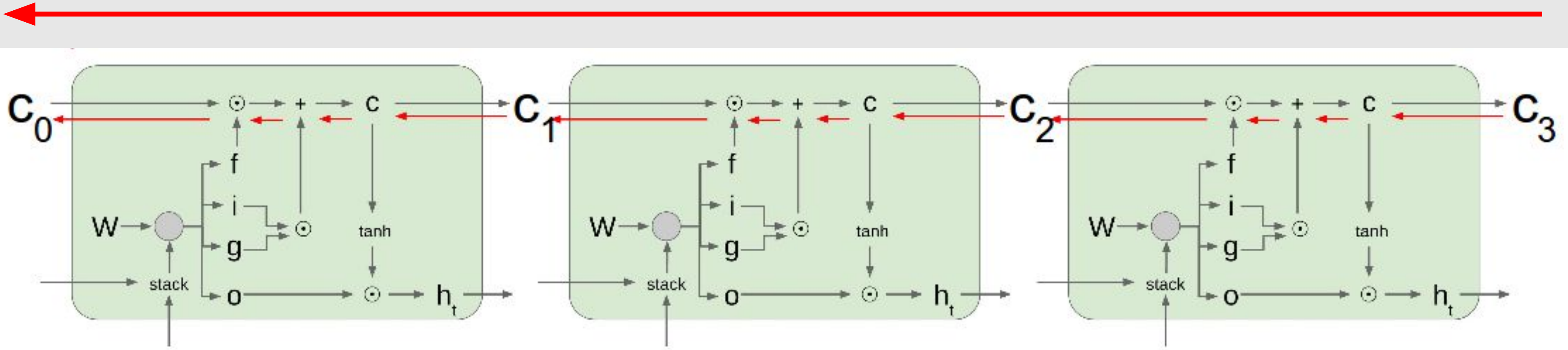
$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \ + \ b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \ + \ b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \ + \ b_o\right)$$



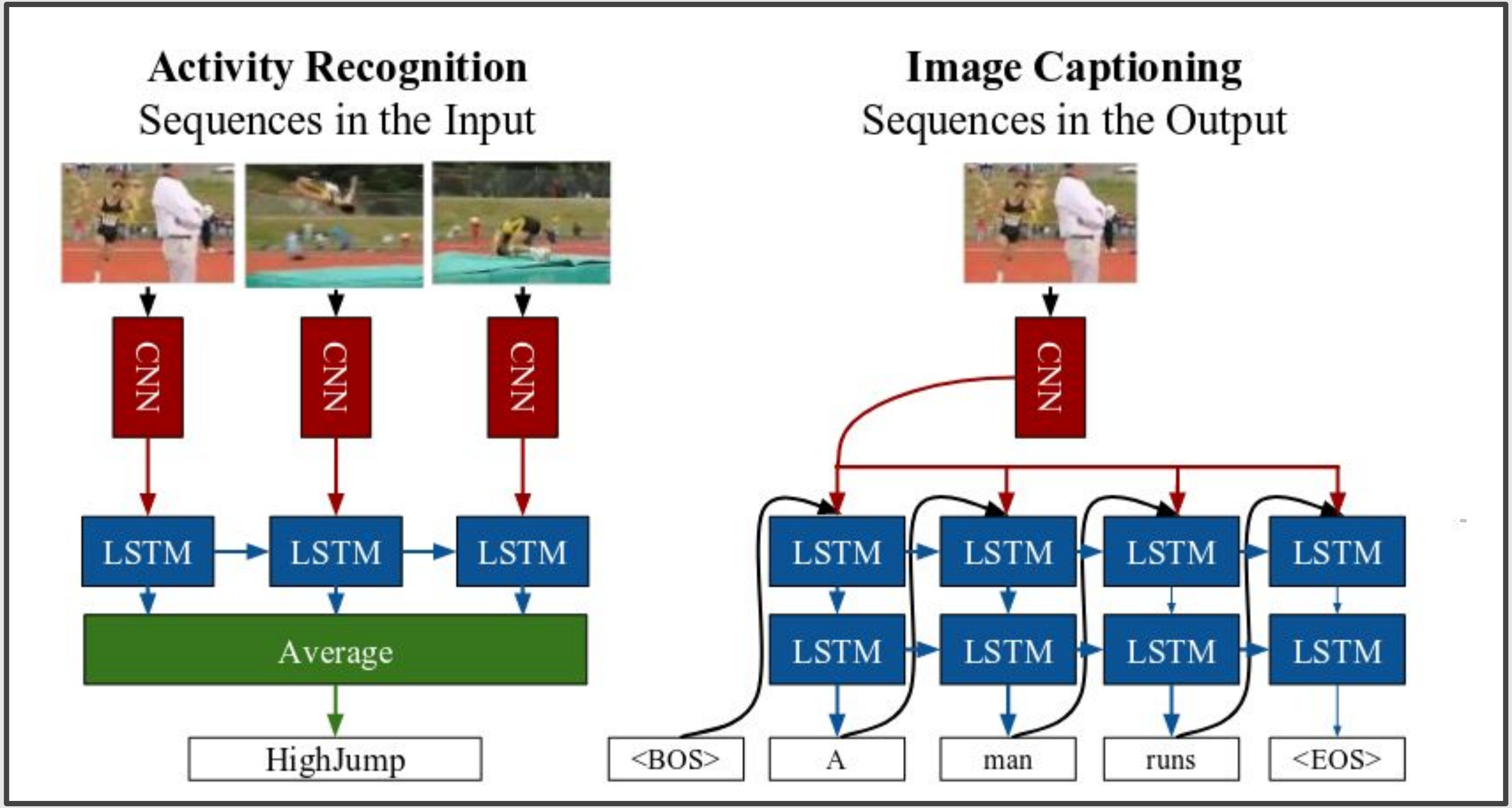
$$C_t = f_t * C_{t-1} + \boldsymbol{(1 - f_t)} * \tilde{C}_t$$

Long Short Term Memory (LSTM)

Uninterrupted gradient flow!



Hochreiter and Schmidhuber, “Long Short Term Memory”, Neural Computation, 1997



Donahue et al., “Long-term Recurrent Convolutional Networks for Visual Recognition and Description”

COCOQA 33827
What is the color of the cat?
Ground truth: black
IMG+BOW: black (0.55)
2-VIS+LSTM: black (0.73)
BOW: gray (0.40)

COCOQA 33827a
What is the color of the couch?
Ground truth: red
IMG+BOW: red (0.65)
2-VIS+LSTM: black (0.44)
BOW: red (0.39)

DAQUAR 1522
How many chairs are there?
Ground truth: two
IMG+BOW: four (0.24)
2-VIS+BLSTM: one (0.29)
LSTM: four (0.19)

DAQUAR 1520
How many shelves are there?
Ground truth: three
IMG+BOW: three (0.25)
2-VIS+BLSTM: two (0.48)
LSTM: two (0.21)

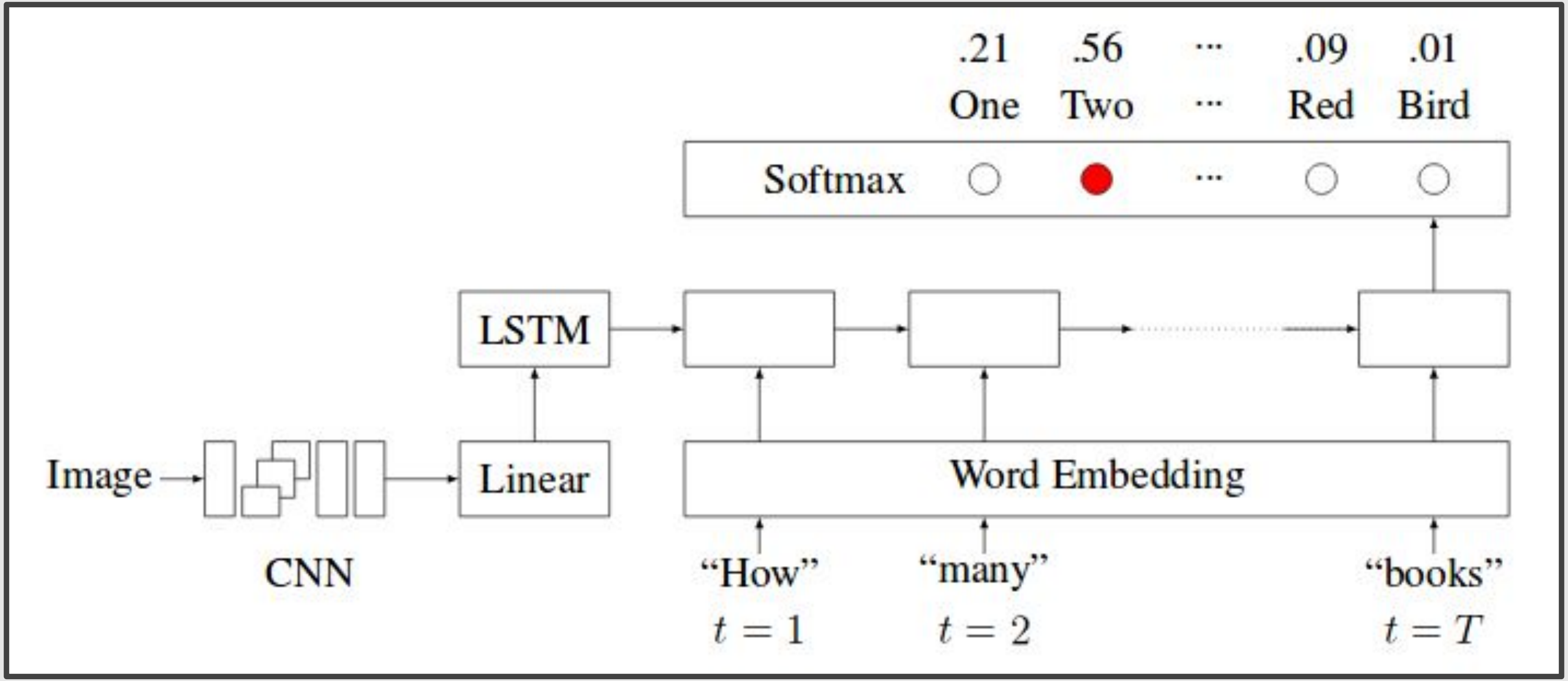
COCOQA 14855
Where are the ripe bananas sitting?
Ground truth: basket
IMG+BOW: basket (0.97)
2-VIS+BLSTM: basket (0.58)
BOW: bowl (0.48)

COCOQA 14855a
What are in the basket?
Ground truth: bananas
IMG+BOW: bananas (0.98)
2-VIS+BLSTM: bananas (0.68)
BOW: bananas (0.14)

DAQUAR 585
What is the object on the chair?
Ground truth: pillow
IMG+BOW: clothes (0.37)
2-VIS+BLSTM: pillow (0.65)
LSTM: clothes (0.40)

DAQUAR 585a
Where is the pillow found?
Ground truth: chair
IMG+BOW: bed (0.13)
2-VIS+BLSTM: chair (0.17)
LSTM: cabinet (0.79)

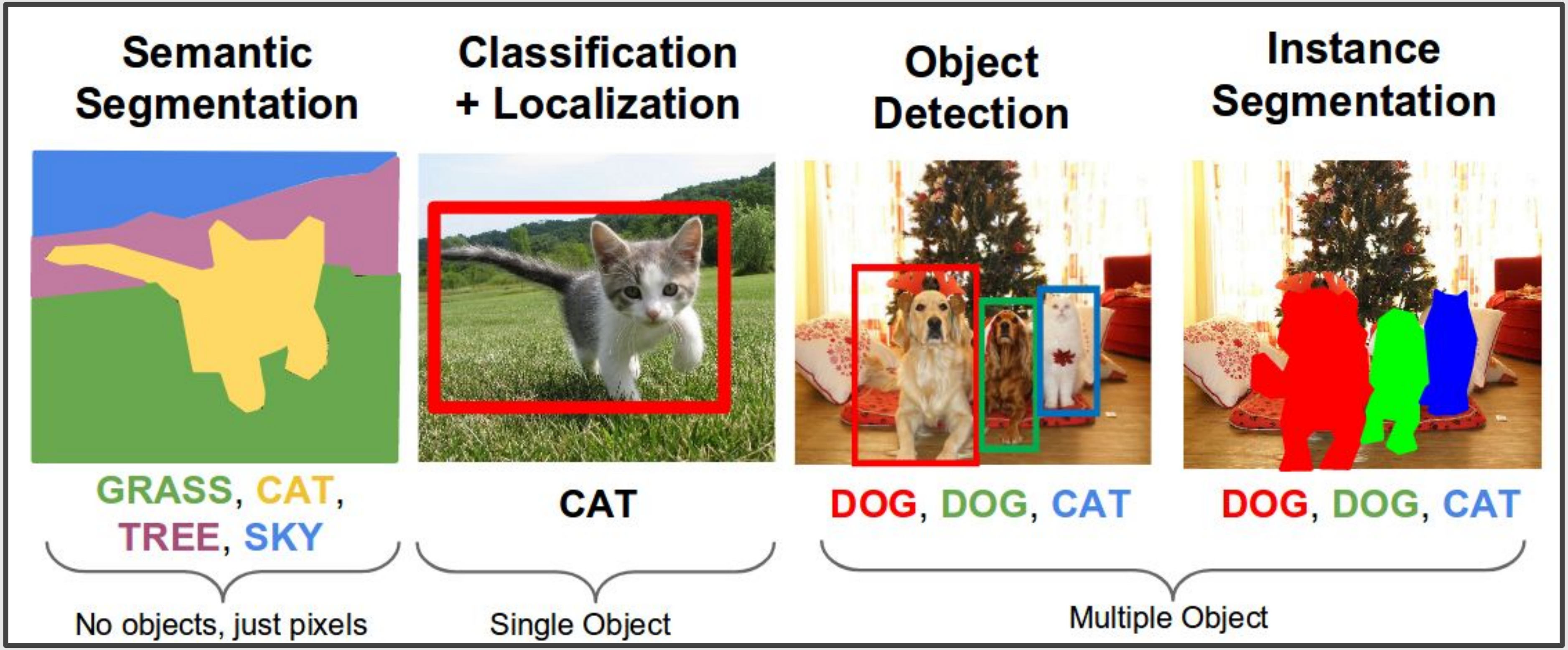
Ren et al., “Exploring Models and Data for Image Question Answering”



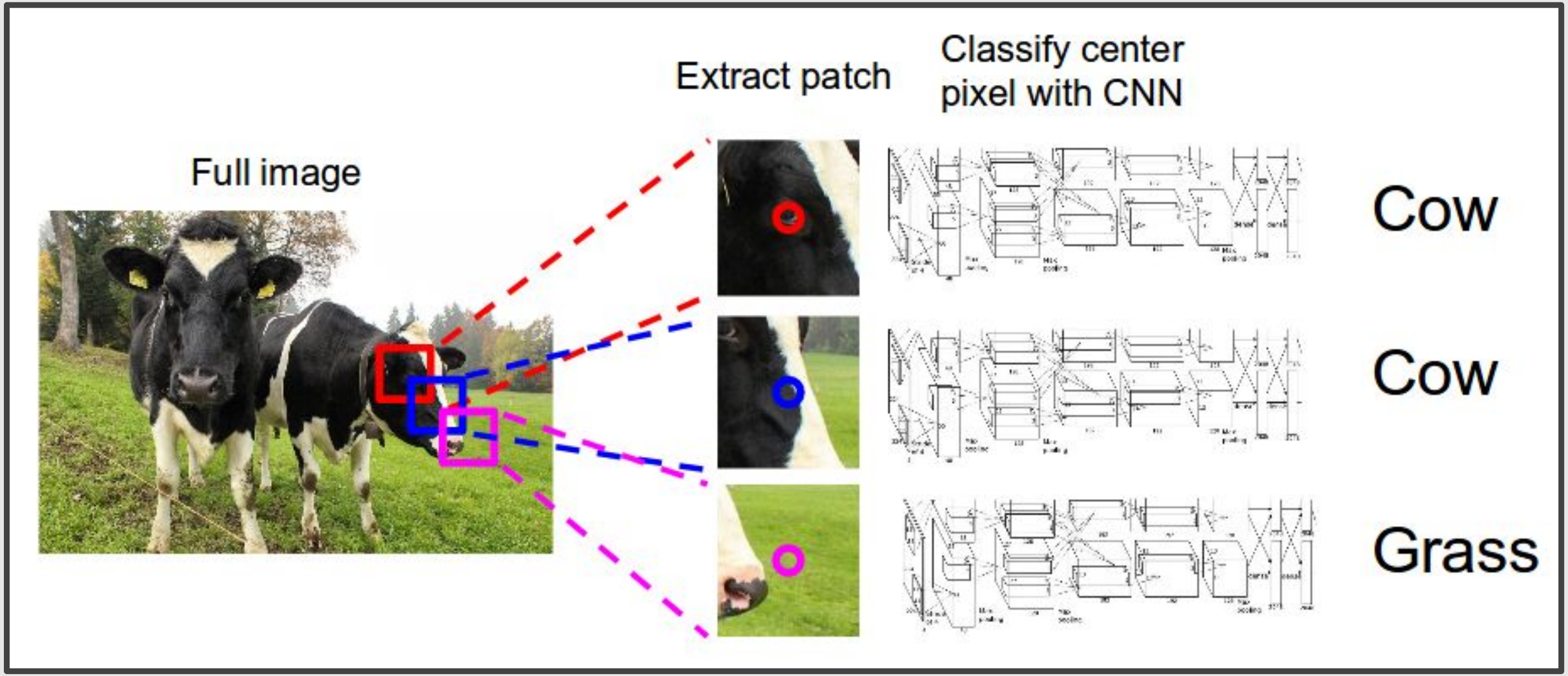
Ren et al., “Exploring Models and Data for Image Question Answering”

Other Tasks …

Other Tasks …

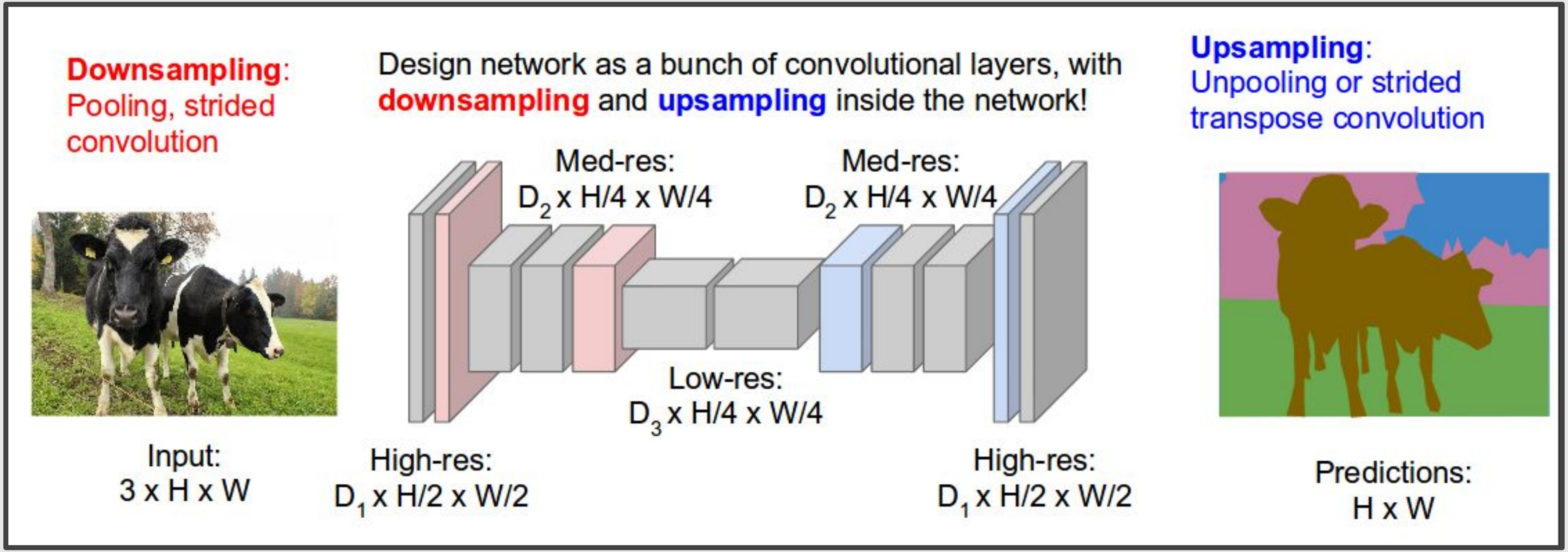

Semantic Segmentation
Classification + Localization
Object Detection
Instance Segmentation
GRASS, CAT, TREE, SKY
CAT
DOG, DOG, CAT
DOG, DOG, CAT
No objects, just pixels
Single Object
Multiple Object

Semantic Segmentation Idea: Sliding Window



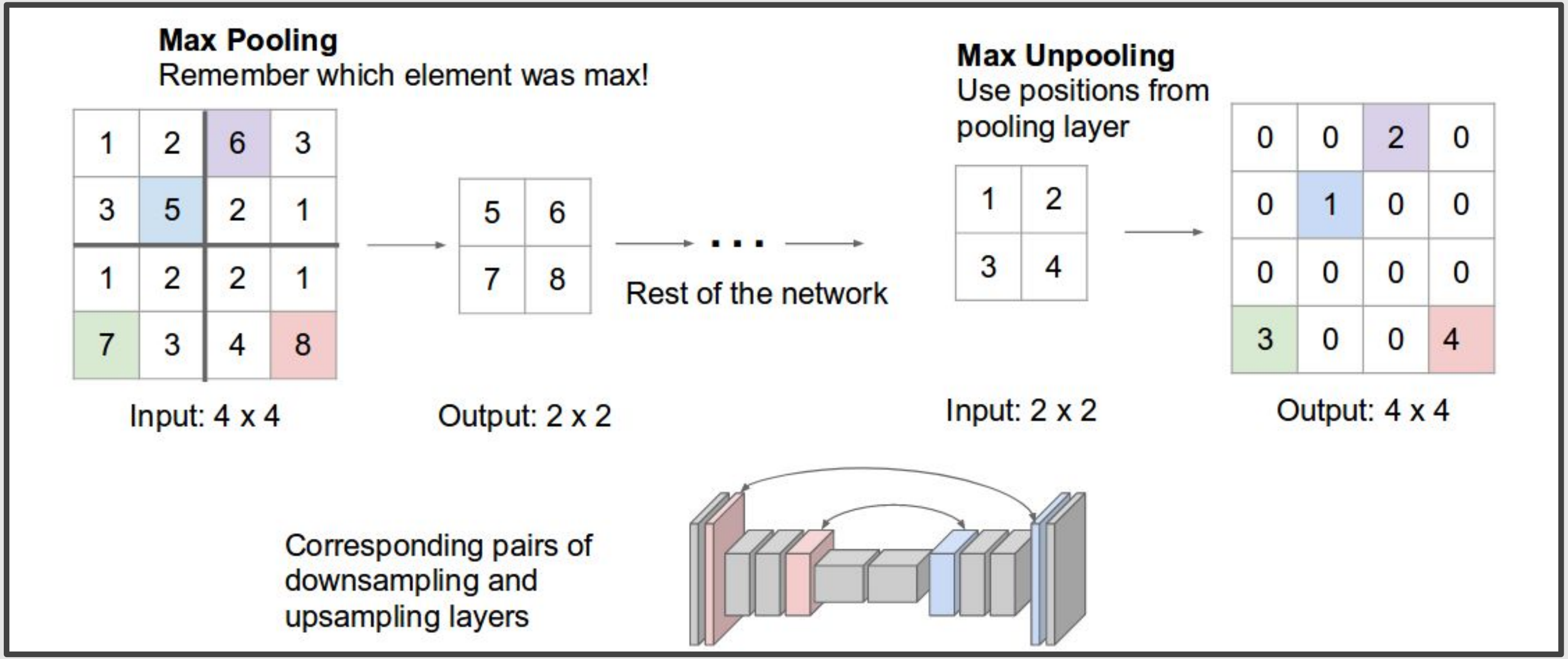
Farabet et al, “Learning Hierarchical Features for Scene Labeling,” TPAMI 2013
Pinheiro and Collobert, “Recurrent Convolutional Neural Networks for Scene Labeling”, ICML 2014

Semantic Segmentation Idea: Fully Convolutional



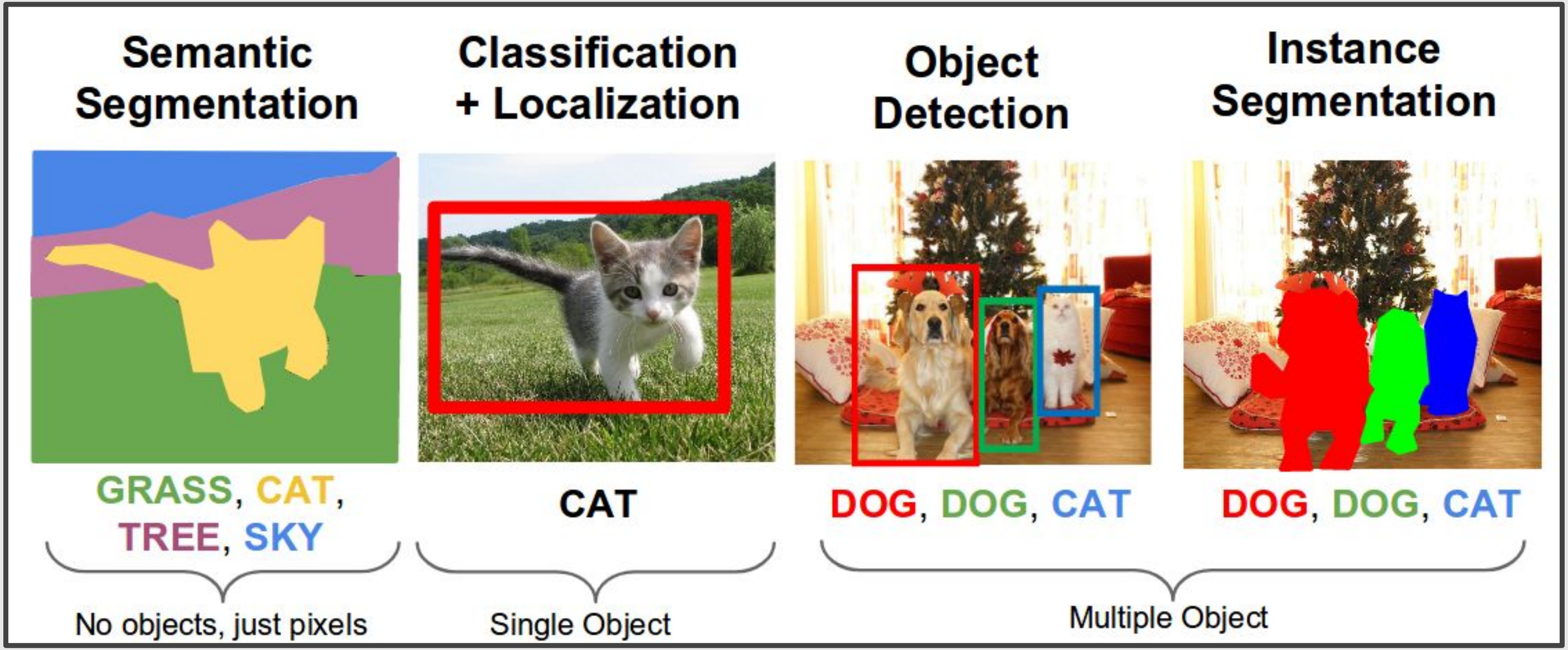
Long, Shelhamer, and Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015
Noh et al, “Learning Deconvolution Network for Semantic Segmentation”, ICCV 2015

Semantic Segmentation Idea: Fully Convolutional

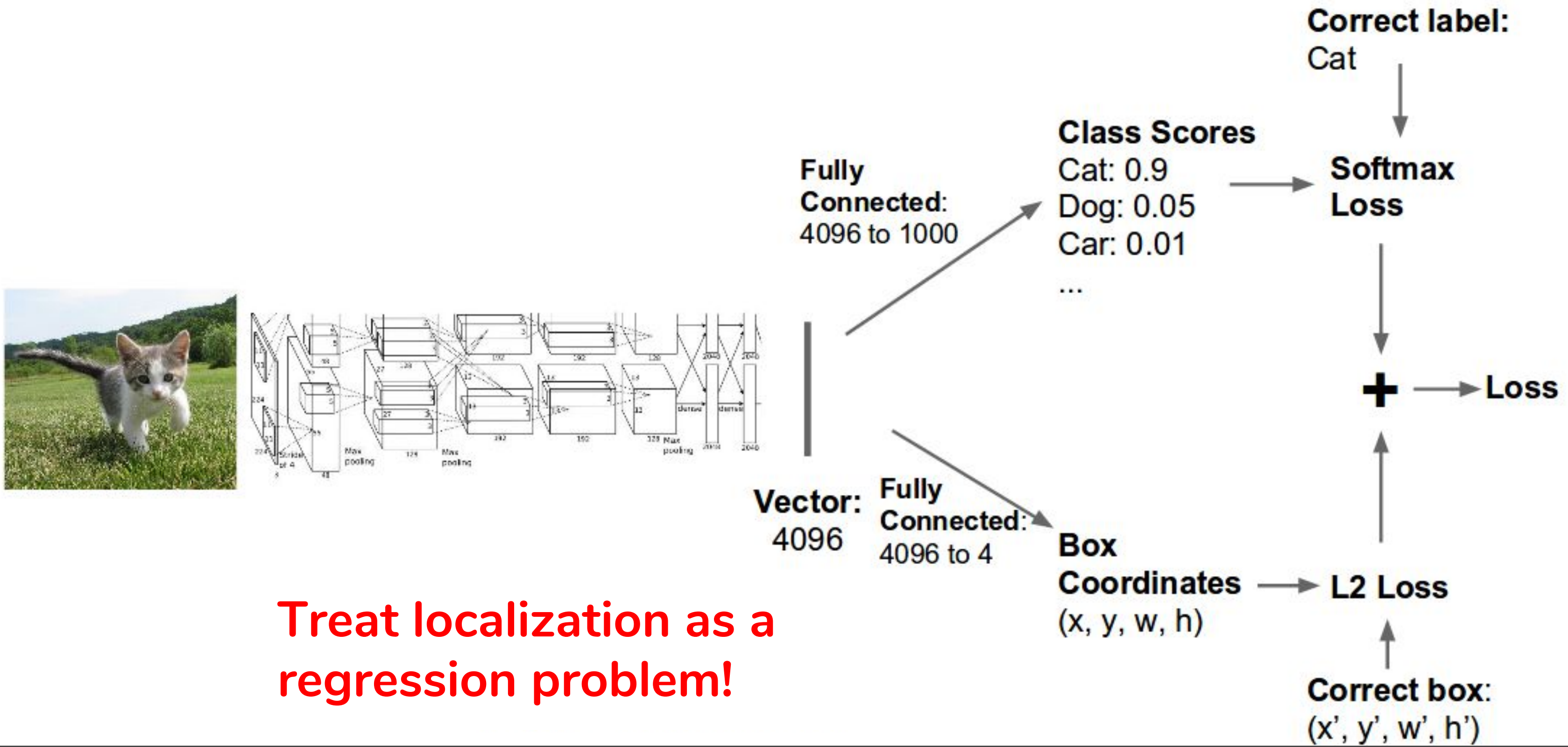


Long, Shelhamer, and Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015
Noh et al, “Learning Deconvolution Network for Semantic Segmentation”, ICCV 2015

Other Tasks …

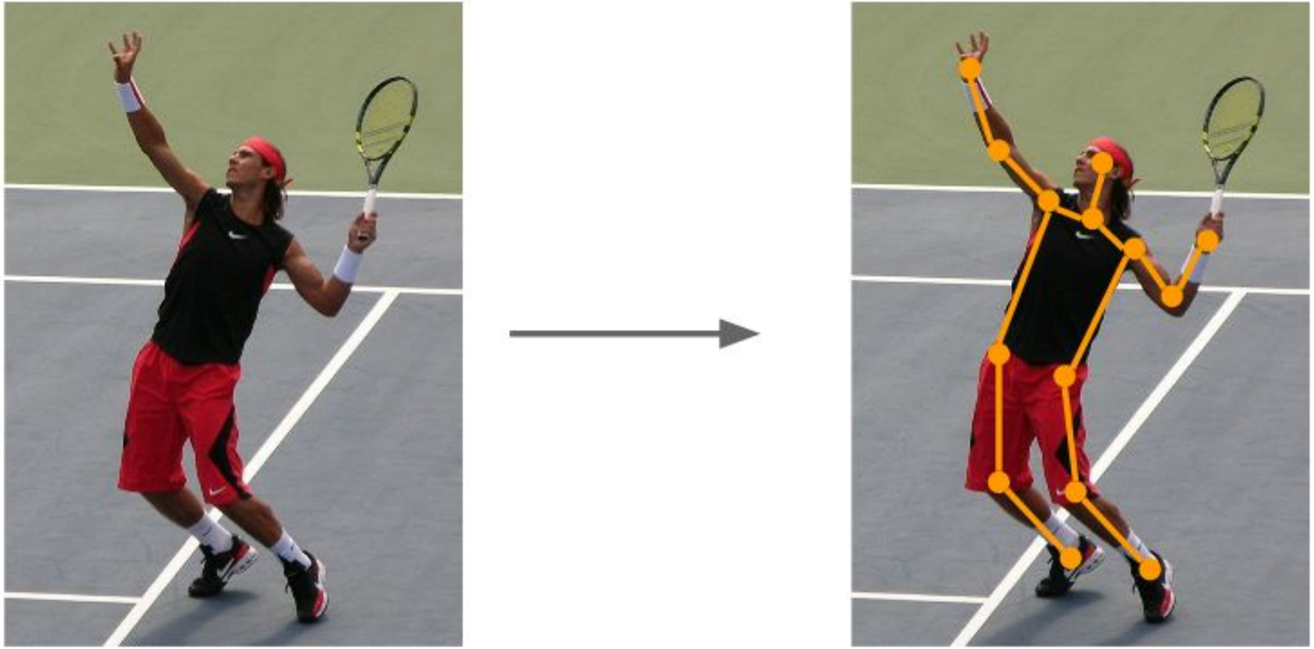

Semantic Segmentation
Classification + Localization
Object Detection
Instance Segmentation
GRASS, CAT, TREE, SKY
CAT
DOG, DOG, CAT
DOG, DOG, CAT
No objects, just pixels
Single Object
Multiple Object

Classification + Localization



Treat localization as a regression problem!

Human Pose Estimation

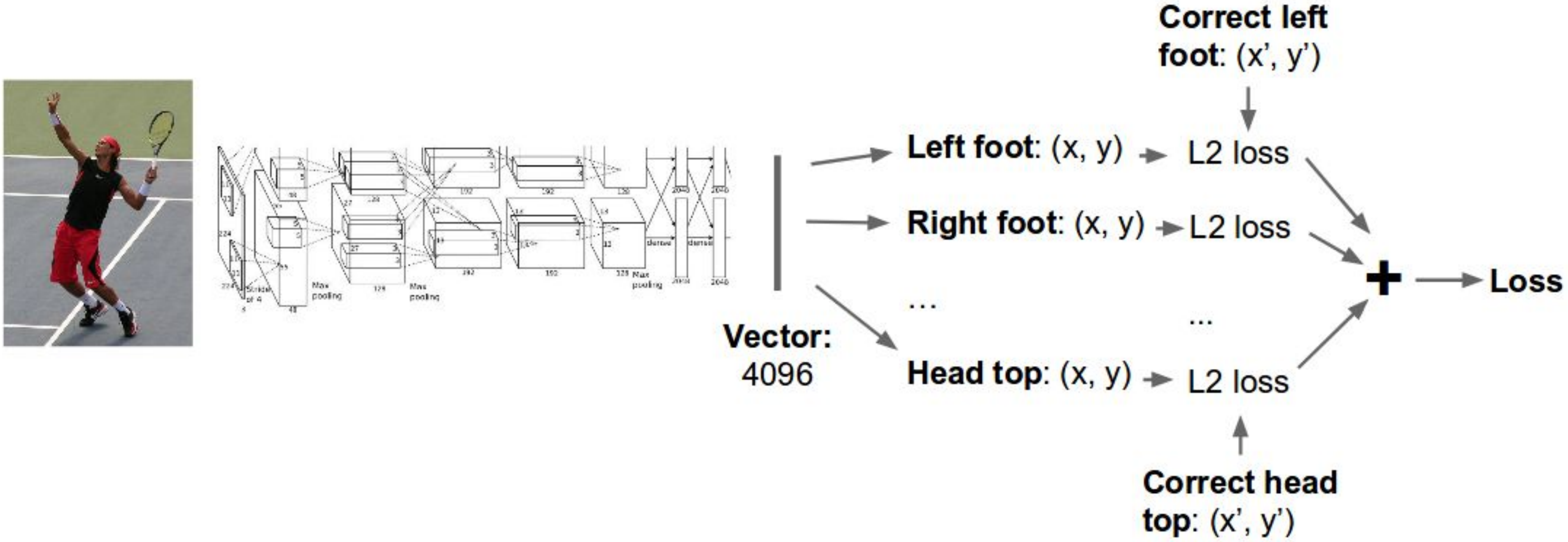


Represent pose as a set of 14 joint positions:

Left / right foot
Left / right knee
Left / right hip
Left / right shoulder
Left / right elbow
Left / right hand
Neck
Head top

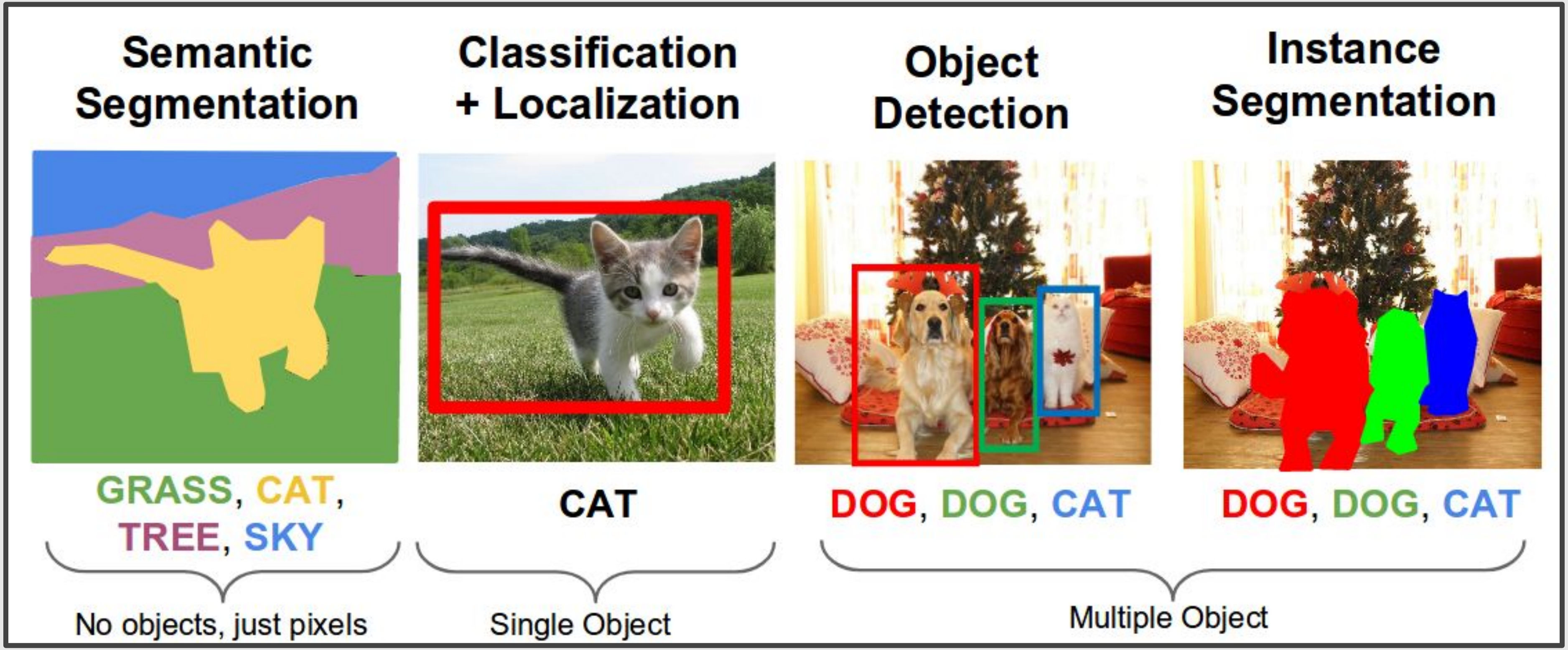
Toshev and Szegedy, “DeepPose: Human Pose Estimation via Deep Neural Networks”, CVPR 2014

Human Pose Estimation



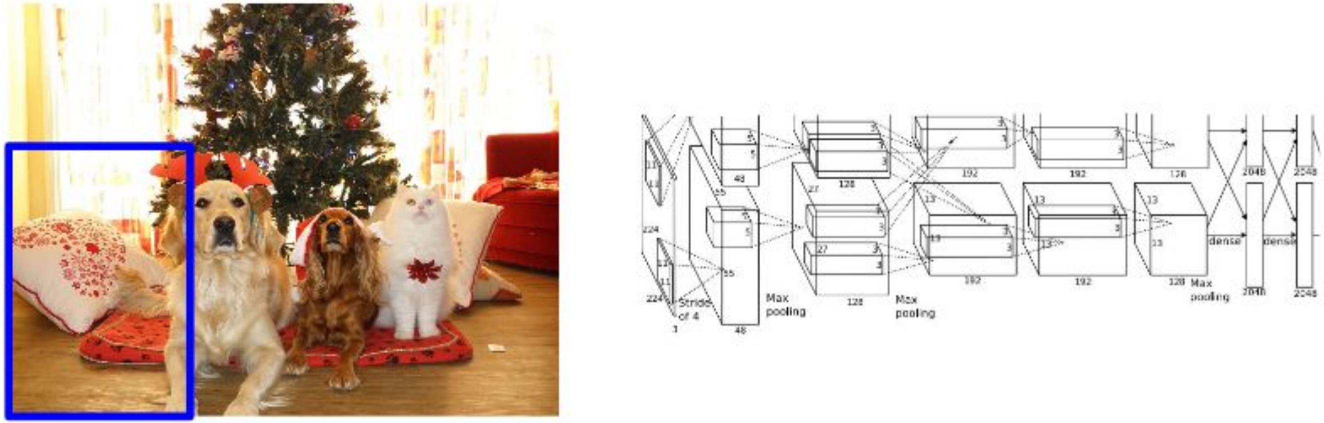
Toshev and Szegedy, “DeepPose: Human Pose Estimation via Deep Neural Networks”, CVPR 2014

Other Tasks …



Semantic Segmentation
Classification + Localization
Object Detection
Instance Segmentation
GRASS, CAT, TREE, SKY
CAT
DOG, DOG, CAT
DOG, DOG, CAT
No objects, just pixels
Single Object
Multiple Object

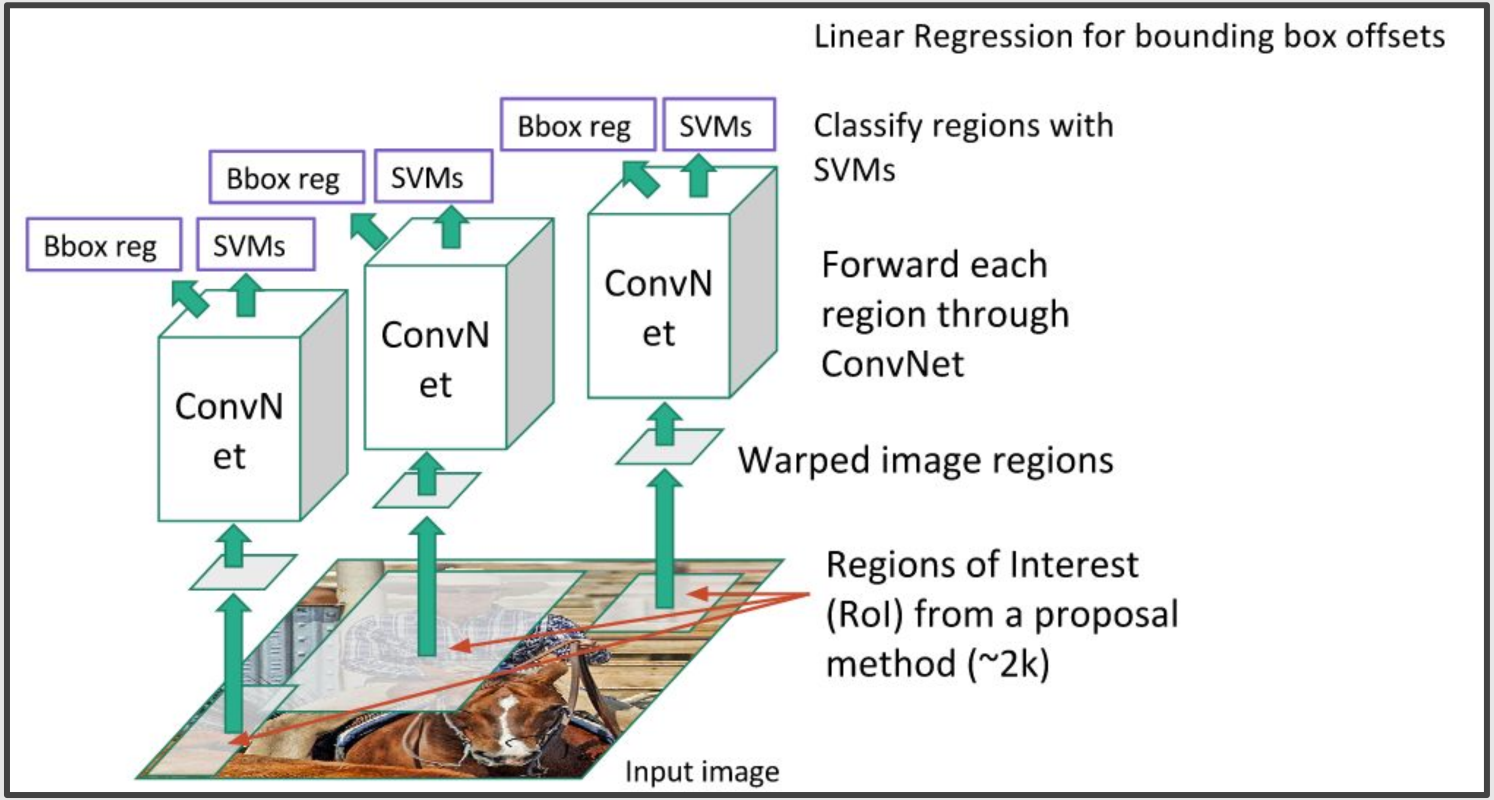
Object Detection as Classification: Sliding Window



Dog? NO
Cat? NO
Background? YES

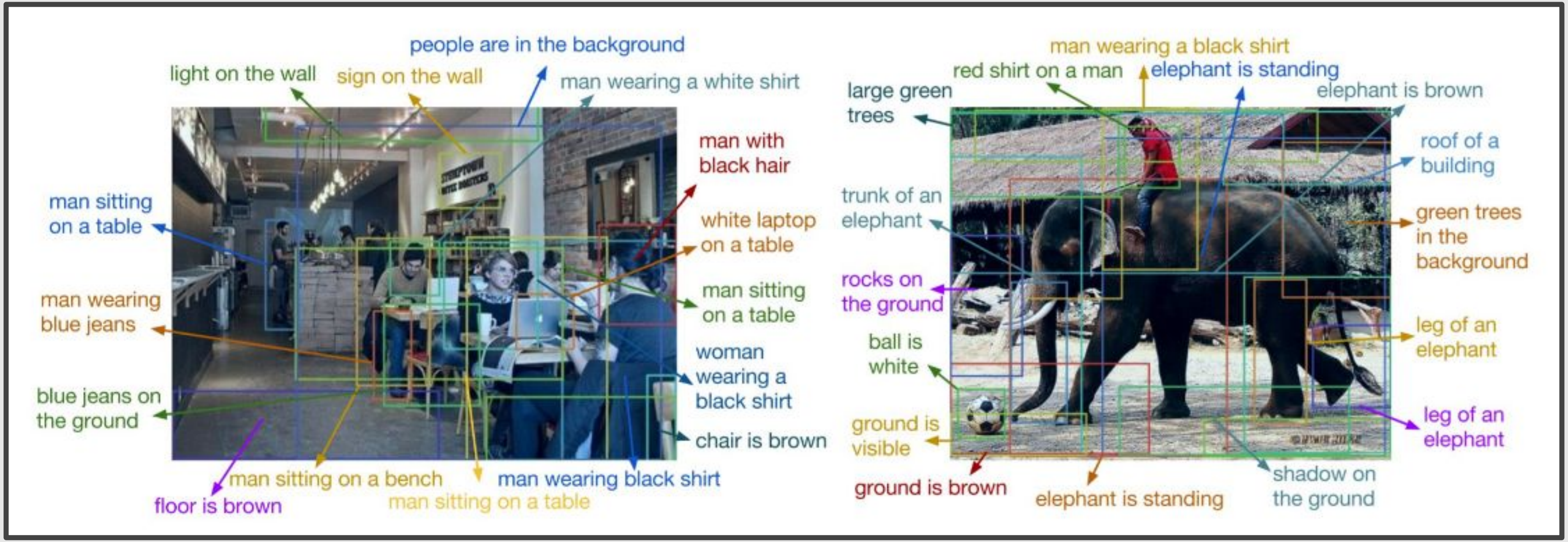
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background

Object Detection: R-CNN



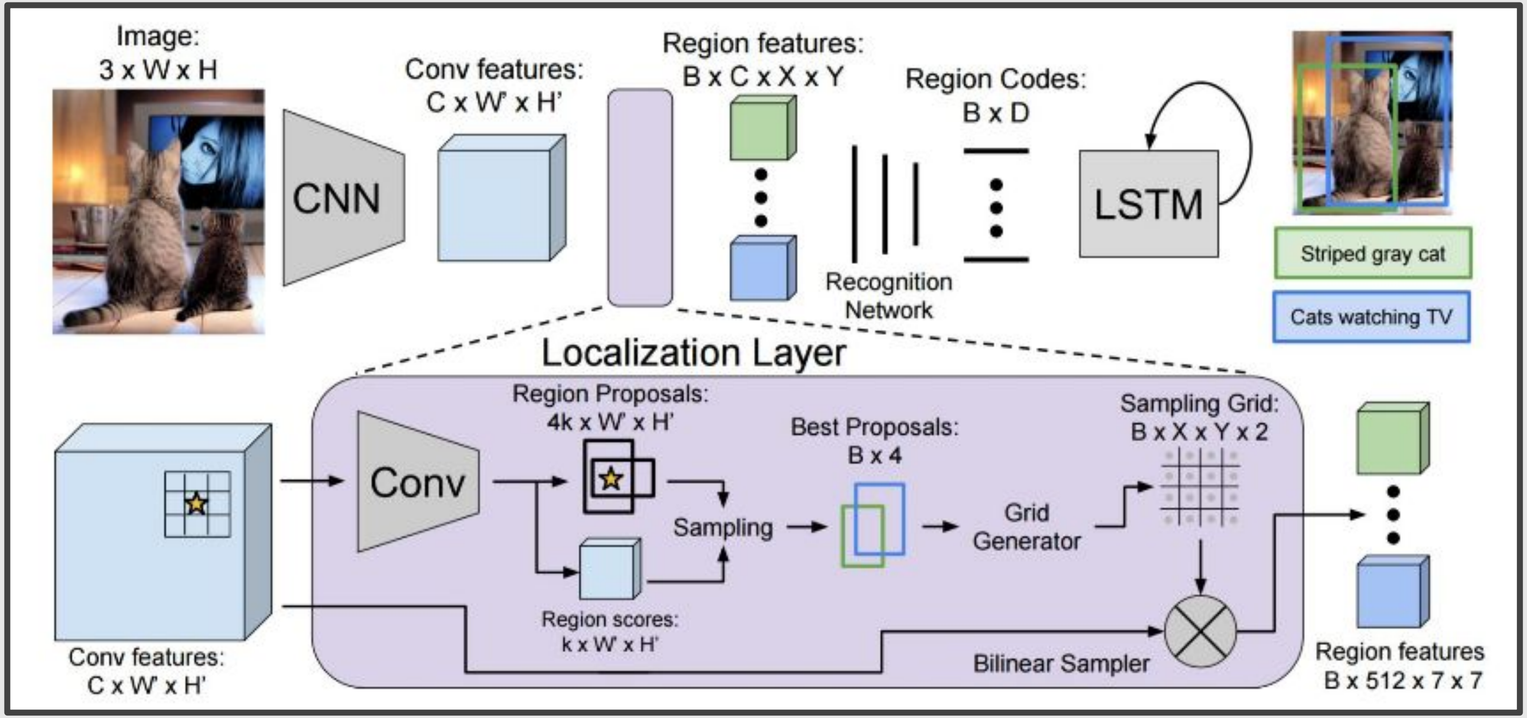
Girshick et al, “Rich feature hierarchies for accurate object detection and semantic segmentation”, CVPR 2014.

Object Detection + Captioning = Dense Captioning



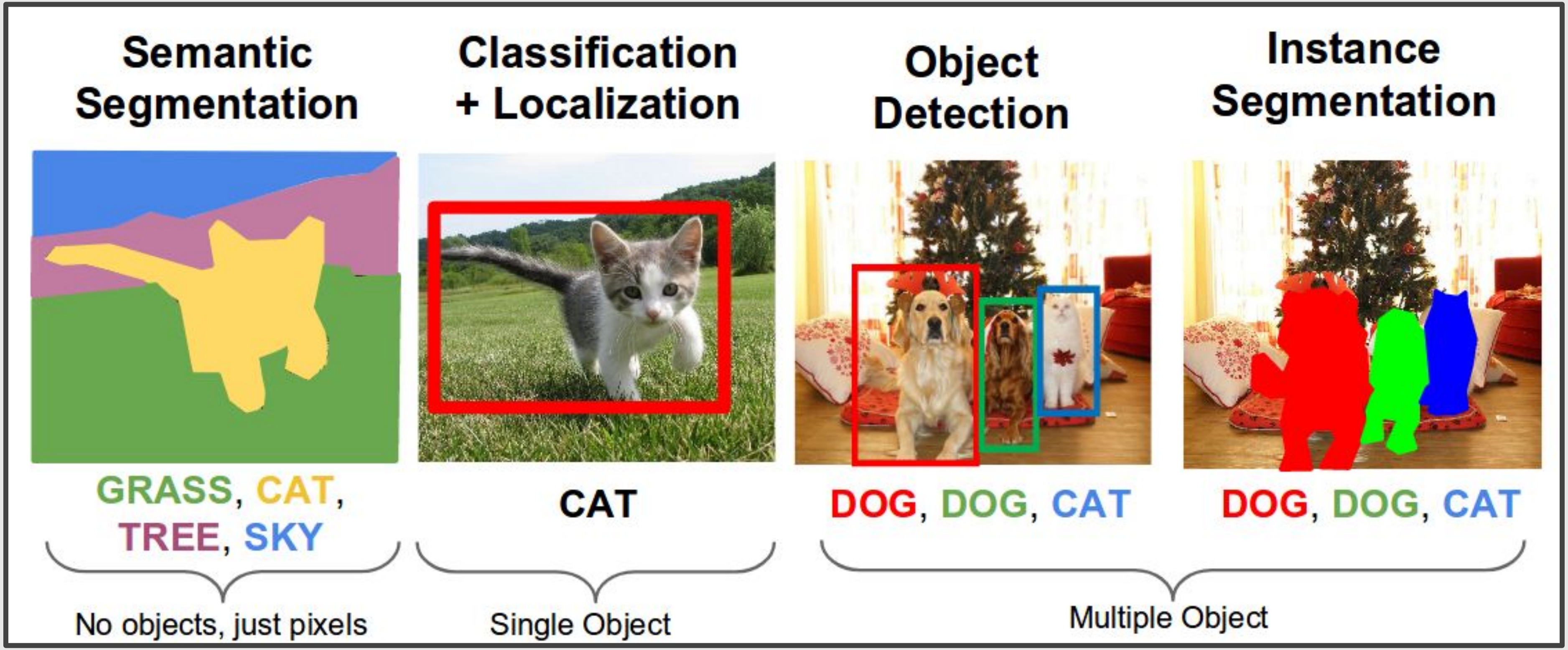
Johnson, Karpathy, and Fei-Fei, “DenseCap: Fully Convolutional Localization Networks for Dense Captioning”, CVPR 2016

Object Detection + Captioning = Dense Captioning



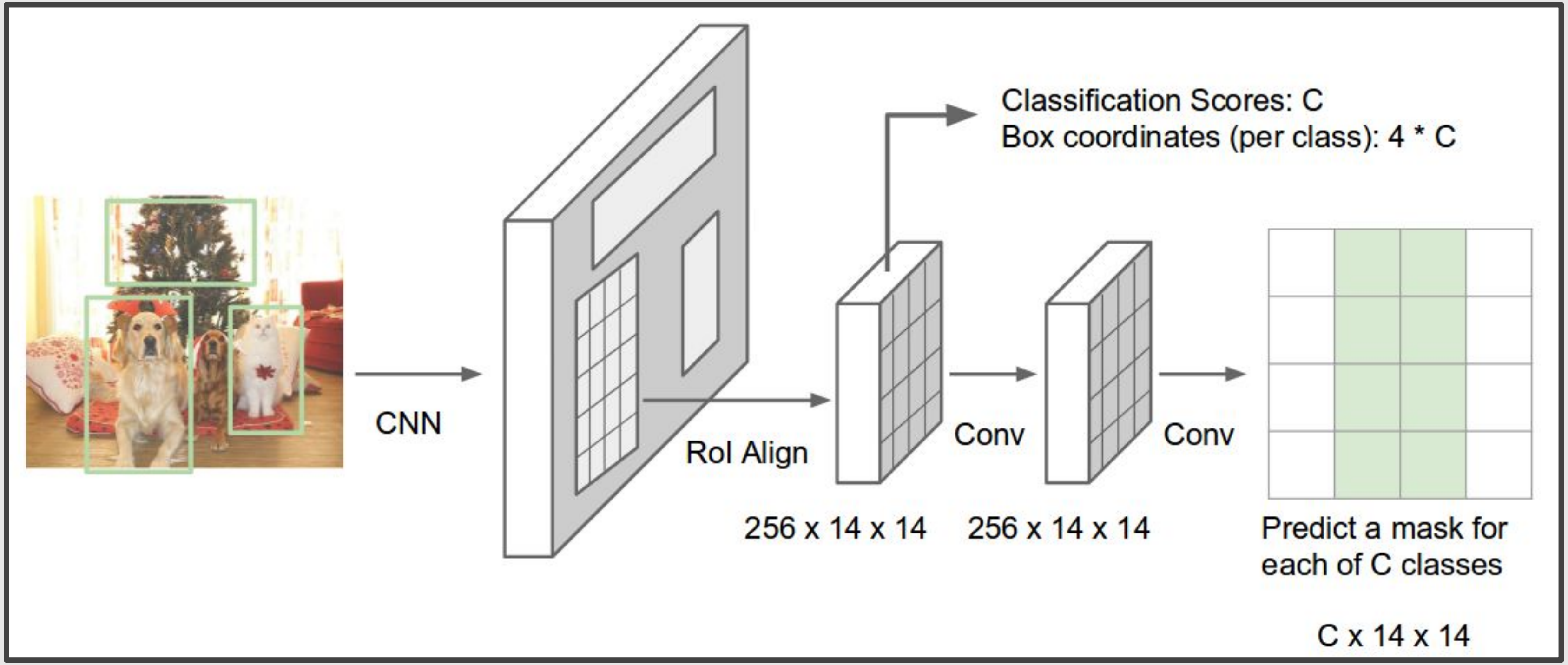
Johnson, Karpathy, and Fei-Fei, “DenseCap: Fully Convolutional Localization Networks for Dense Captioning”, CVPR 2016

Other Tasks …



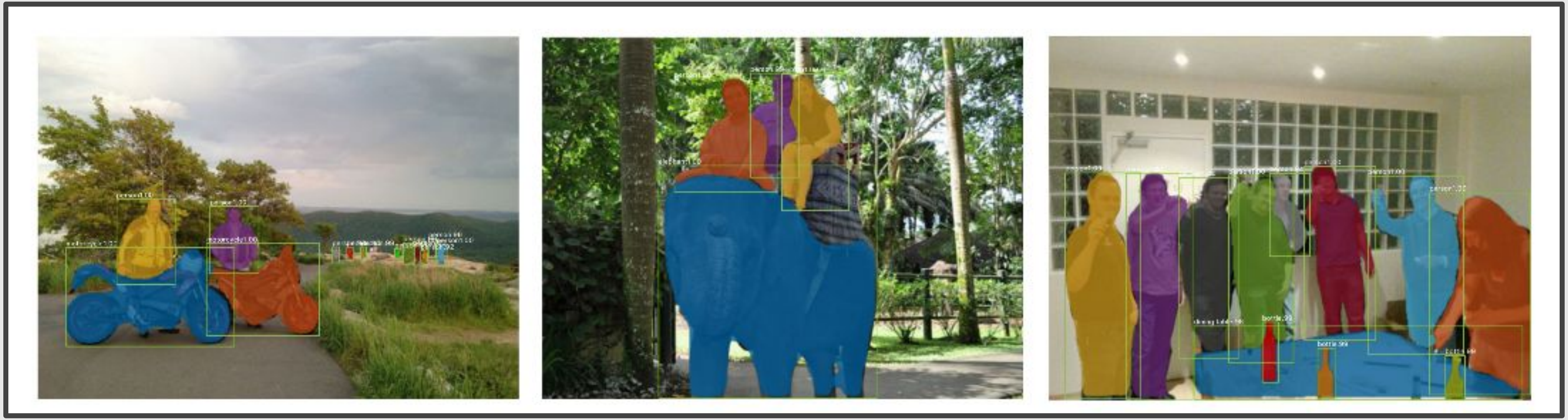
Semantic Segmentation
Classification + Localization
Object Detection
Instance Segmentation
GRASS, CAT, TREE, SKY
CAT
DOG, DOG, CAT
DOG, DOG, CAT
No objects, just pixels
Single Object
Multiple Object

Instance Segmentation



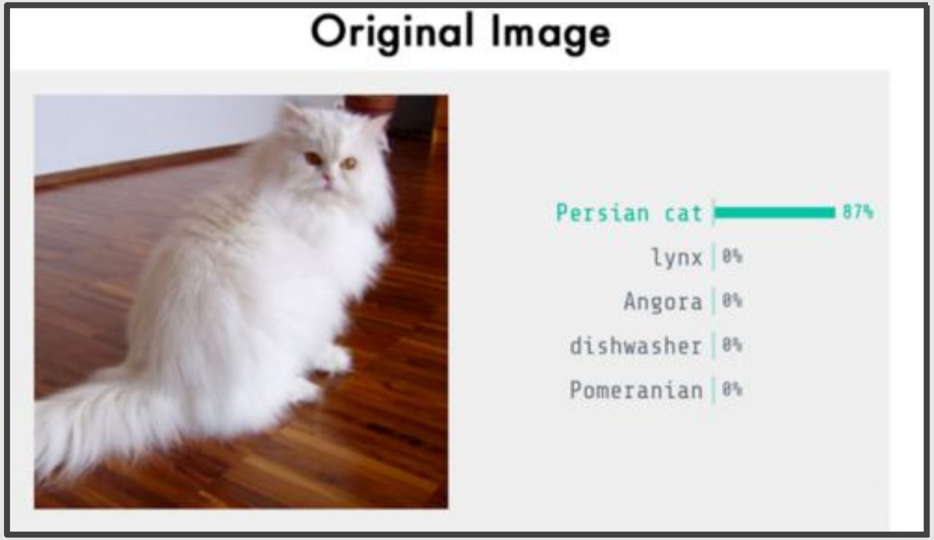
He et al, “Mask R-CNN”, arXiv 2017

Instance Segmentation



He et al, “Mask R-CNN”, arXiv 2017

How to Intentionally Trick Neural Networks



https://medium.com/@ageitgey/machine-learning-is-fun-part-8-how-to-intentionally-trick-neural-networks-b55da32b7196

How to Intentionally Trick Neural Networks

https://medium.com/@ageitgey/machine-learning-is-fun-part-8-how-to-intentionally-trick-neural-networks-b55da32b7196

References

Deep Learning Books

- Deep Learning, http://www.deeplearningbook.org/contents/rnn.html

Deep Learning Courses

- Recurrent Neural Networks - The Math of Intelligence (Week 5): https://youtu.be/BwmddtPFWtA
- LSTM Networks - The Math of Intelligence (Week 8): https://youtu.be/9zhrxE5PQgY
- Understanding LSTM Networks: http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- https://www.coursera.org/learn/neural-network
- CS231n: Convolutional Neural Networks for Visual Recognition: http://cs231n.stanford.edu/
- "The 3 popular courses on Deep Learning": https://medium.com/towards-data-science/the-3-popular-courses-for-deeplearning-ai-ac37d4433bd