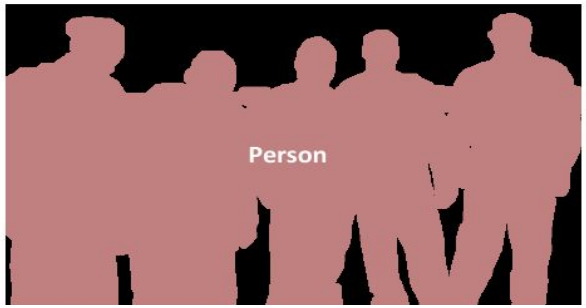
Deep machine learning for Computer Vision

Andrii Liubonko
Samsung R&D Institute Ukraine

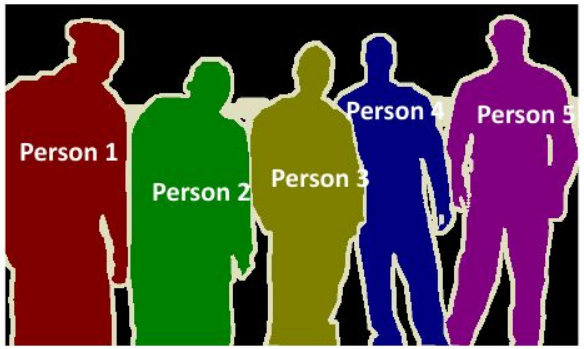
Visual Perception Problems

Classification + Localization



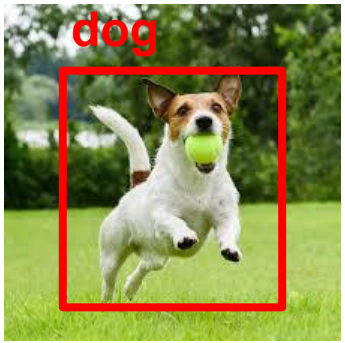
Semantic Segmentation

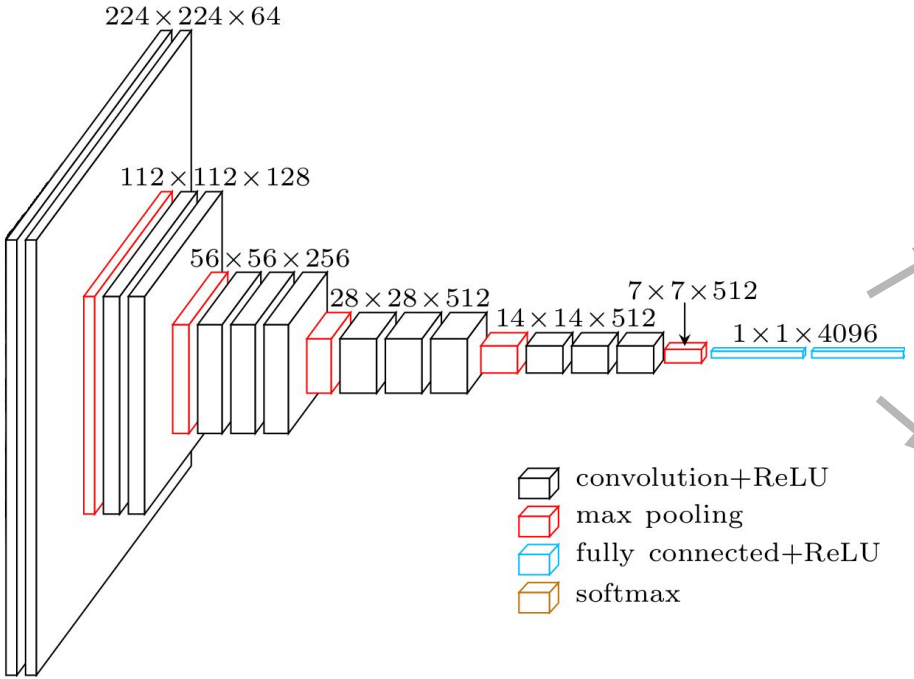
Object Detection



Instance Segmentation

Localization





4096x1000

FC

[Score]
dog: 0.98
cow: 0.01
....

Softmax Loss

+

4096x4

FC

L2 Loss

[BB]:
x,y,h,w

Multihead Network
(Multitask Loss)

Object Detection

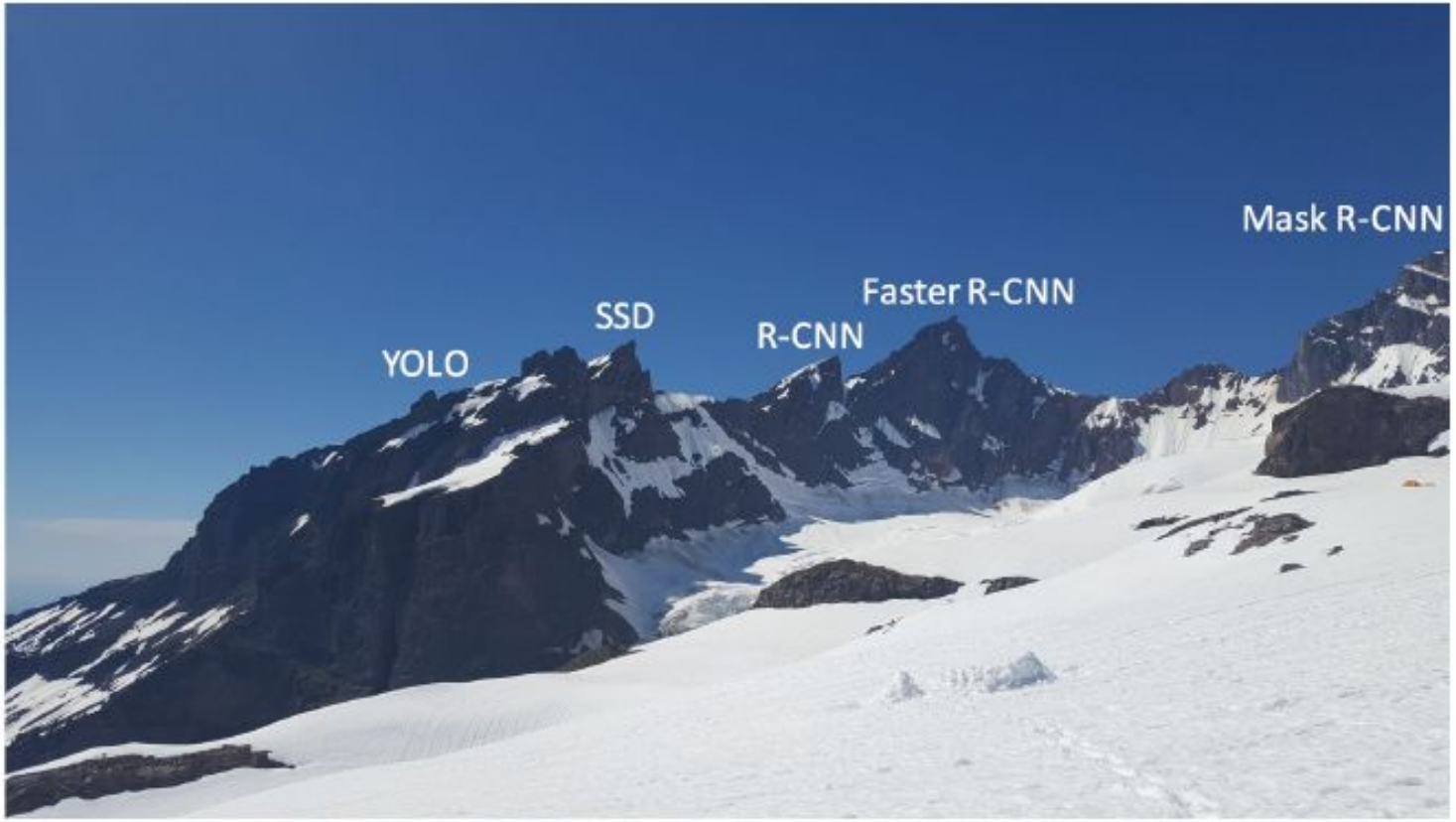
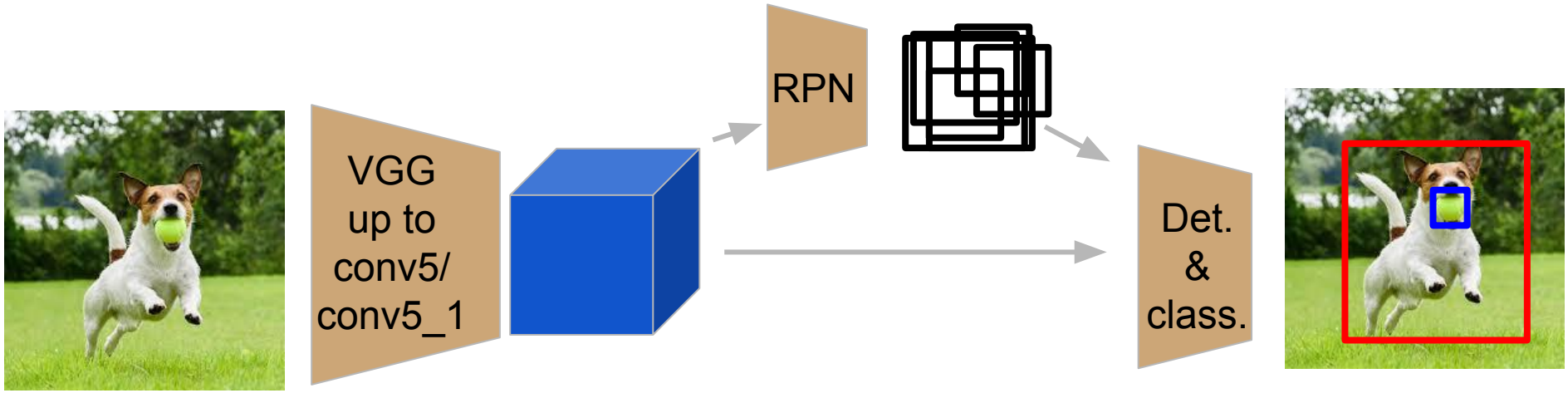
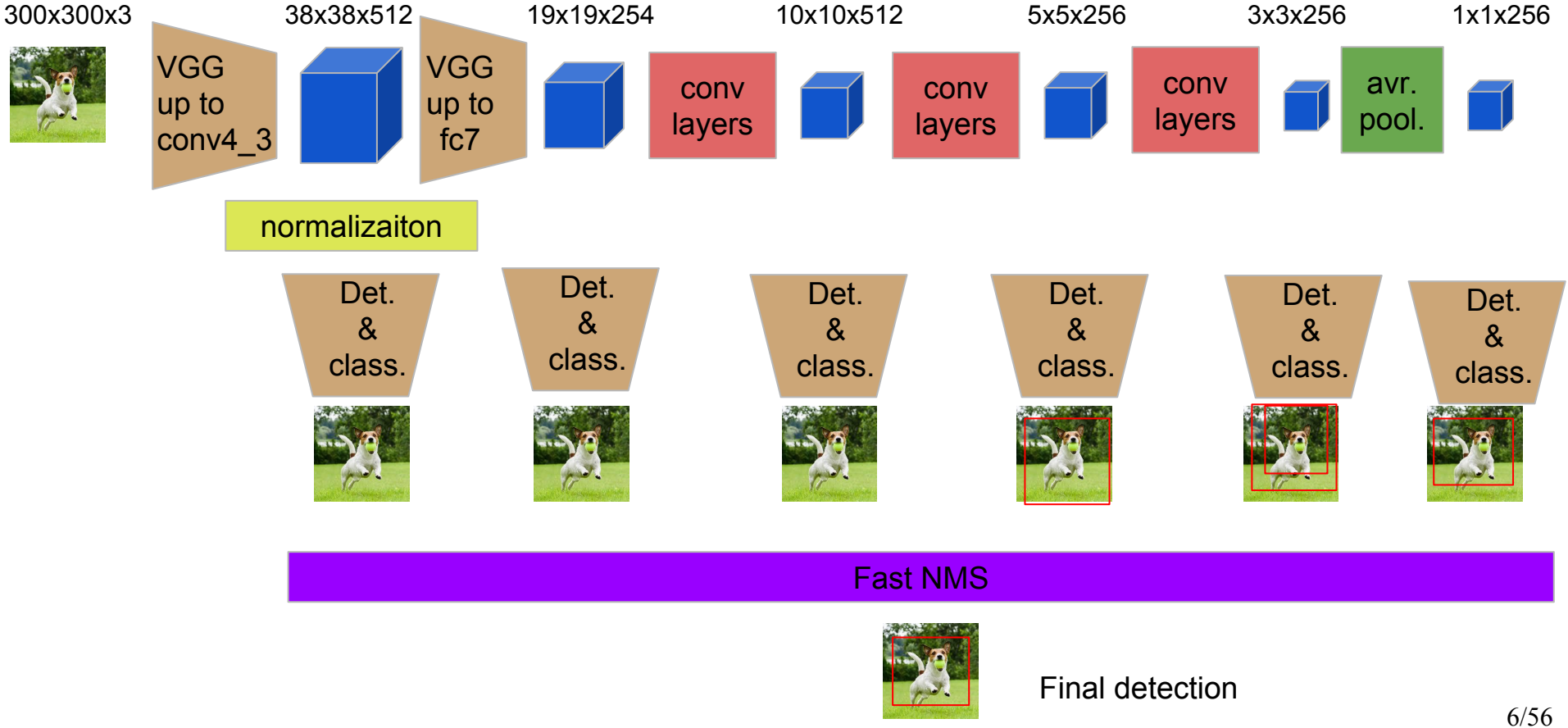


Image from: http://deeplearning.csail.mit.edu/instance_ross.pdf

Faster R-CNN [Overall architecture]



SSD architecture [Overall architecture]

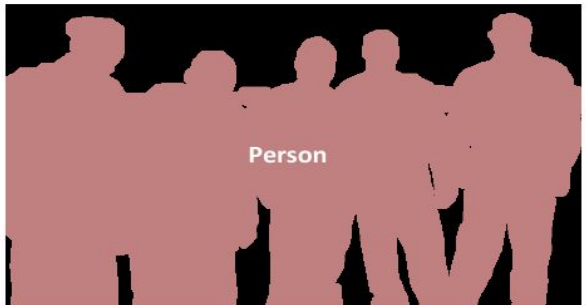


Contents

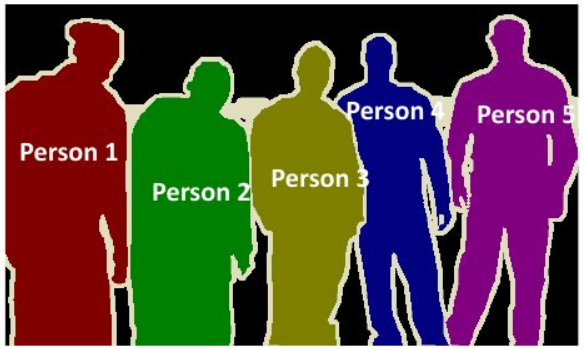
Visual Perception Problems

Classification + Localization

Object Detection

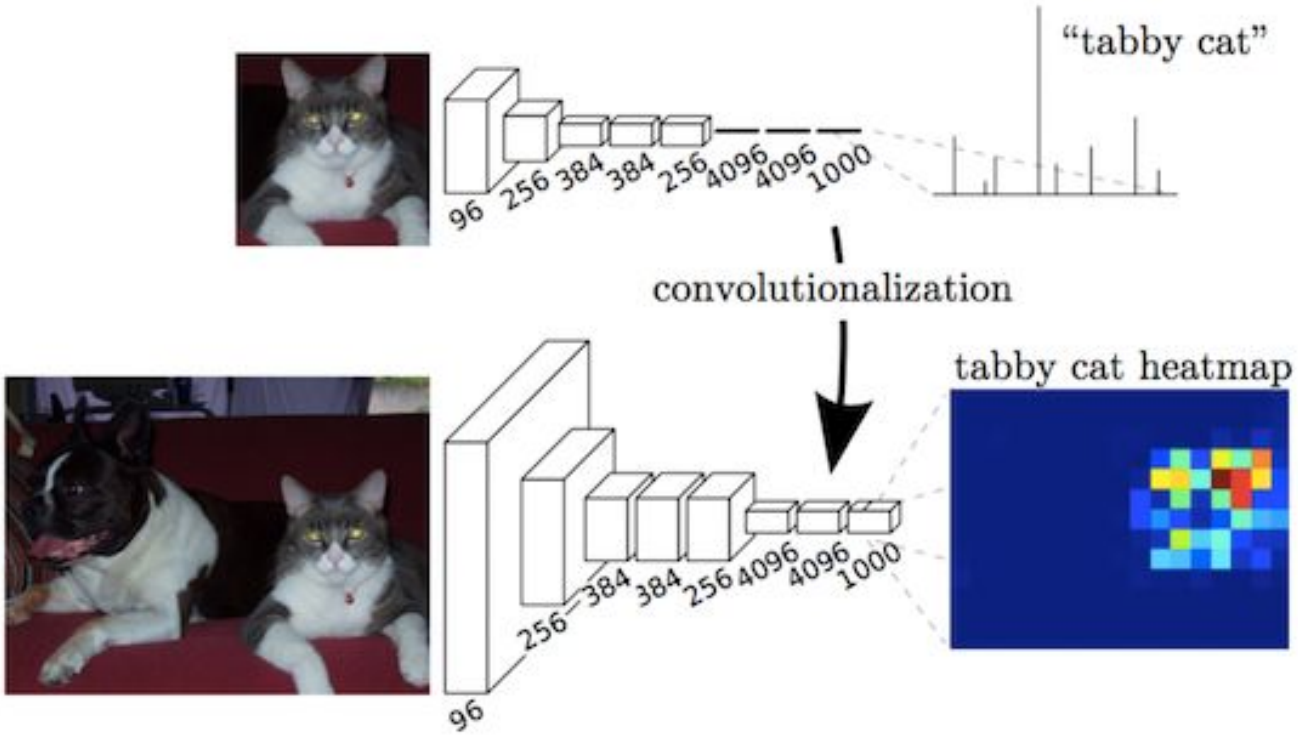


Semantic Segmentation



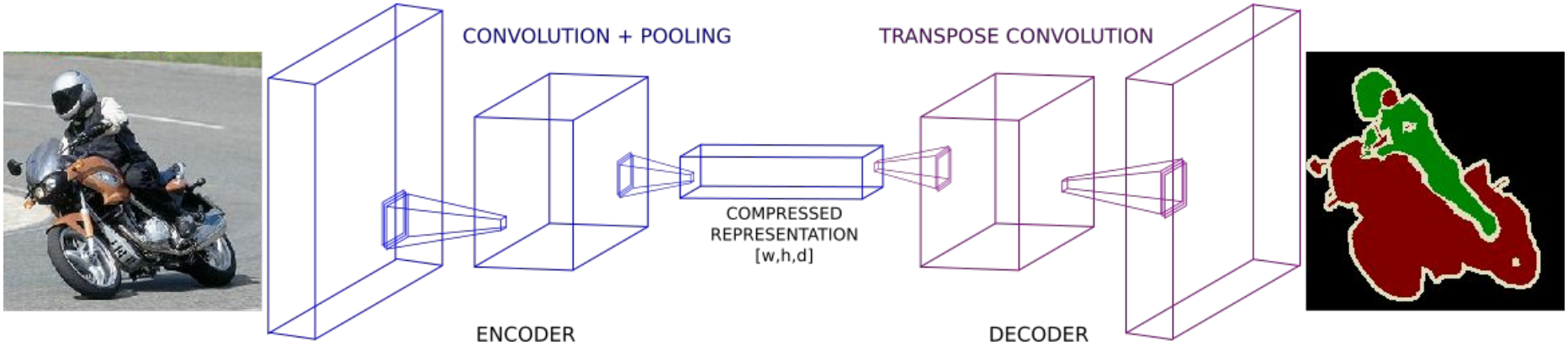
Instance Segmentation

Semantic Segmentation [FCN]



http://blog.qure.ai/notes/semantic-segmentation-deep-learning-review

Semantic Segmentation [SegNet]



http://blog.qure.ai/notes/semantic-segmentation-deep-learning-review

Semantic Segmentation [DeepLab]

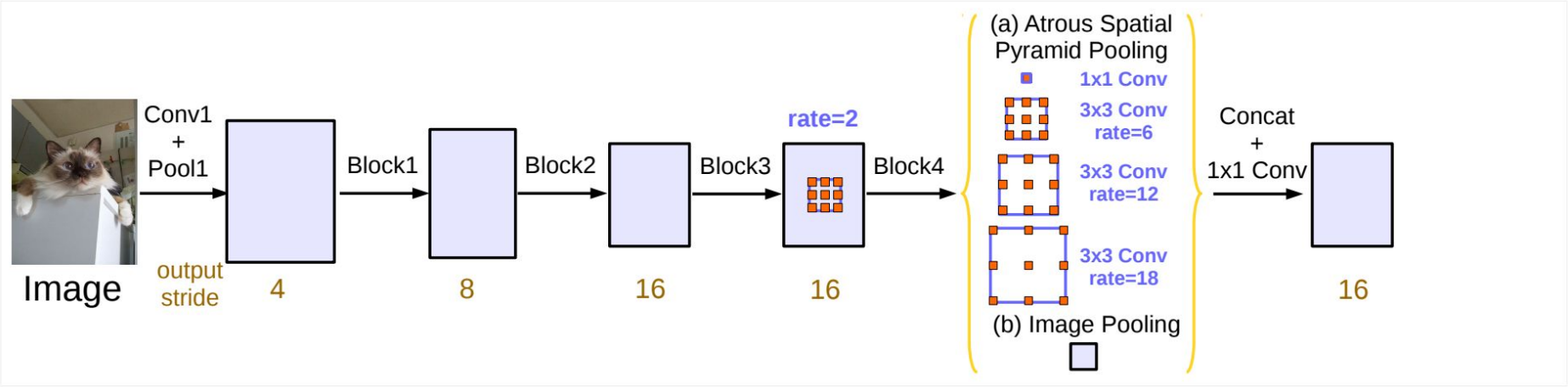
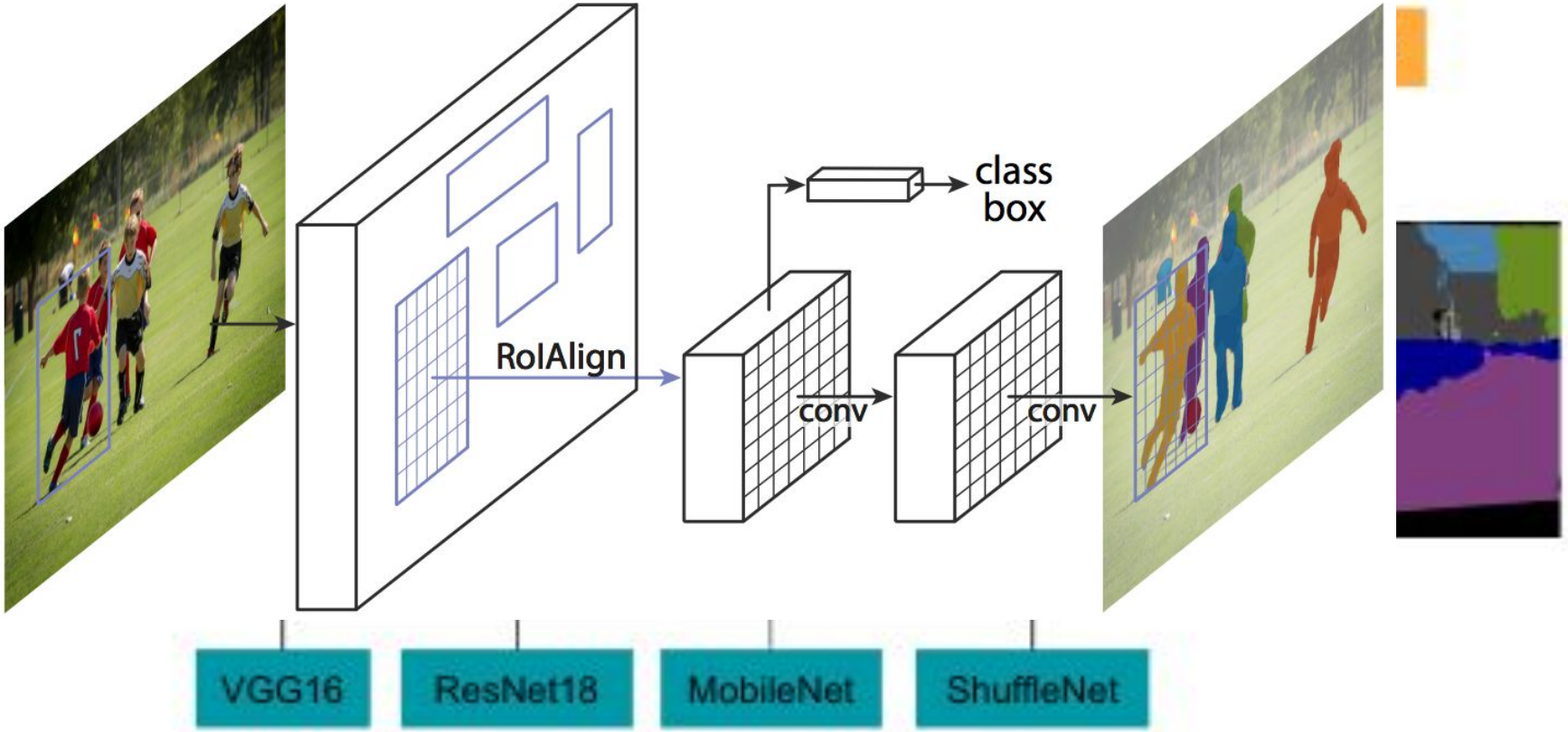


Figure 5. Parallel modules with atrous convolution (ASPP), augmented with image-level features.

arXiv: 1706.05587

Instance Segmentation [Mask-RCNN]



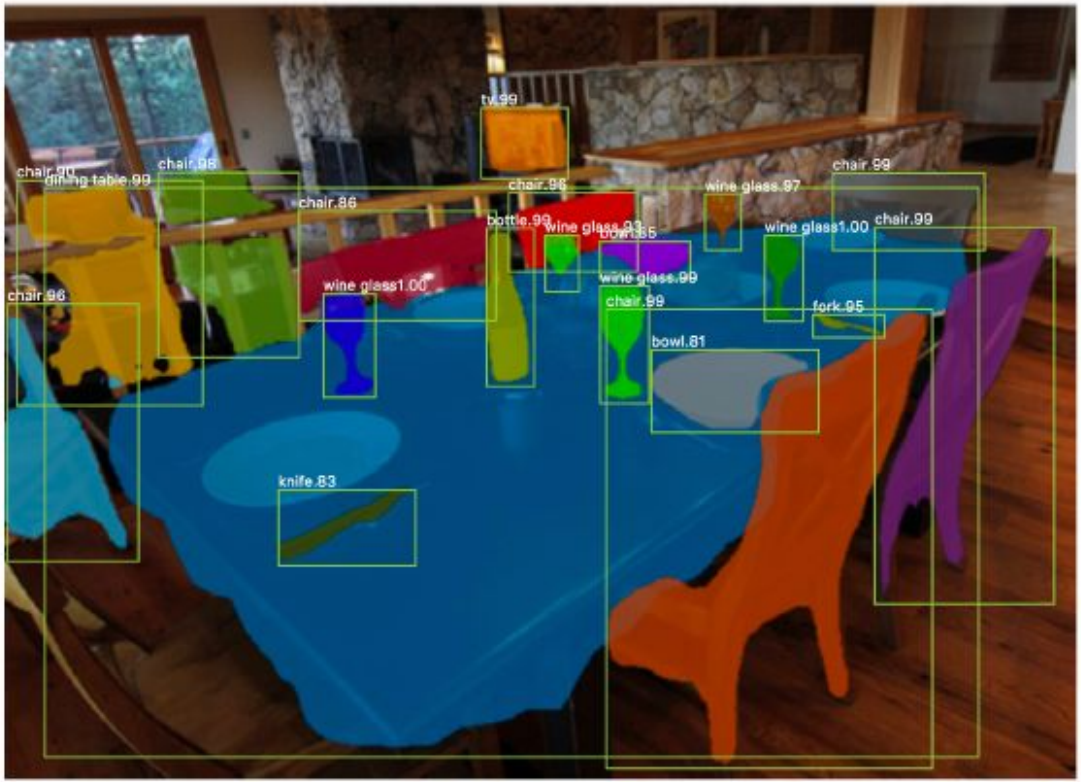
arXiv:1703.06870

Instance Segmentation [Mask-RCNN]

1. Backbone Architecture
2. Scale Invariance (e.g. Feature Pyramid Network (FPN))
3. Region Proposal Network (RPN)
4. Region of interest feature alignment (RoIAlign)
5. Multi-task network head
    a. Box classifier
    b. Box regressor
    c. Mask predictor
    d. Keypoint predictor

From Ross Girshik CVPR 2017 Tutorial

Instance Segmentation



Contents

Metric Learning

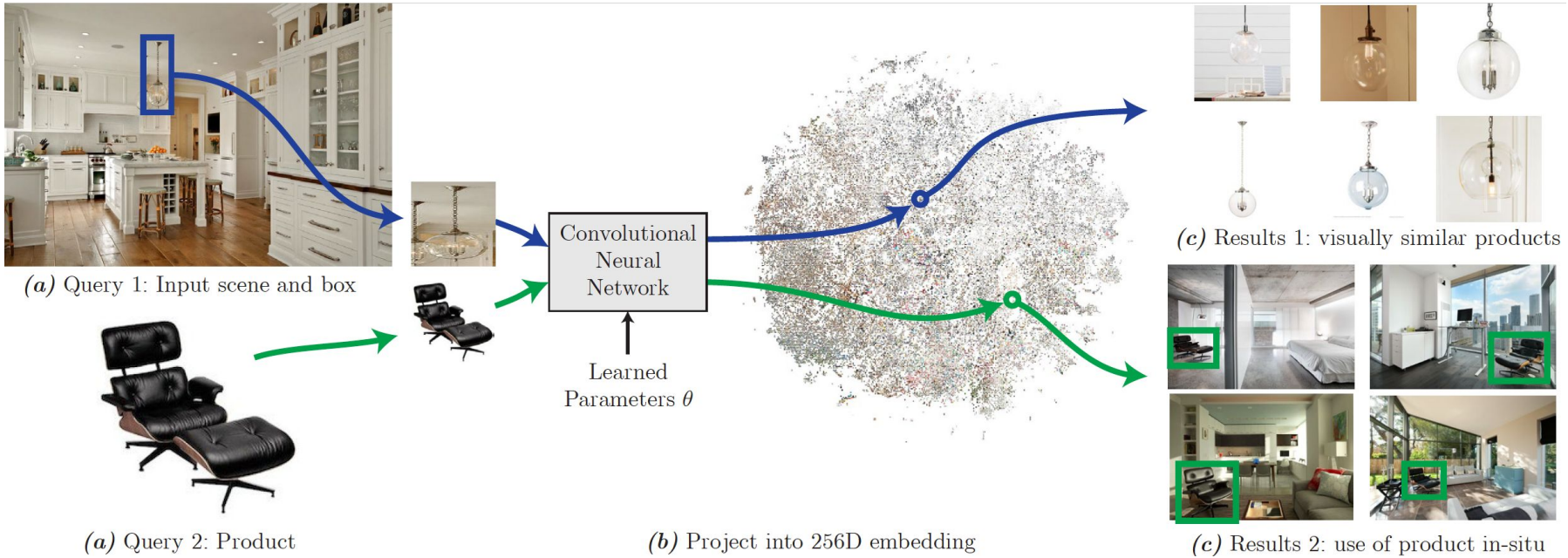
Metric learning (Distance Metric Learning, Similarity Learning)

Goal: to learn a distance function over objects, which measure how similar or related two objects are.

Useful for:

- Image Retrival
- Face Recognition
- Object/Face Re-Identification
- Recommendation Systems

Metric Learning



In deep metric learning, feature embedding function is modeled as a deep neural network. This feature embedding function embeds input images into feature embedding space with a certain desired condition.

Metric Learning [Dataset]

- CUB-200-2011:
  species 200 of birds with 11.788 images
- CARS196: 198 classes of cars and 16.185 images
- Stanford Online Products:
  23.000 classes and 120.000 img.

Metric Learning [Dataset]

The search quality is measured with **recall@K** metric.

Each query image first retrives K nearest neighbours from the test set and receives score 1 if a image of the same class is retrived among the K nearest neighbours and 0 otherwise.

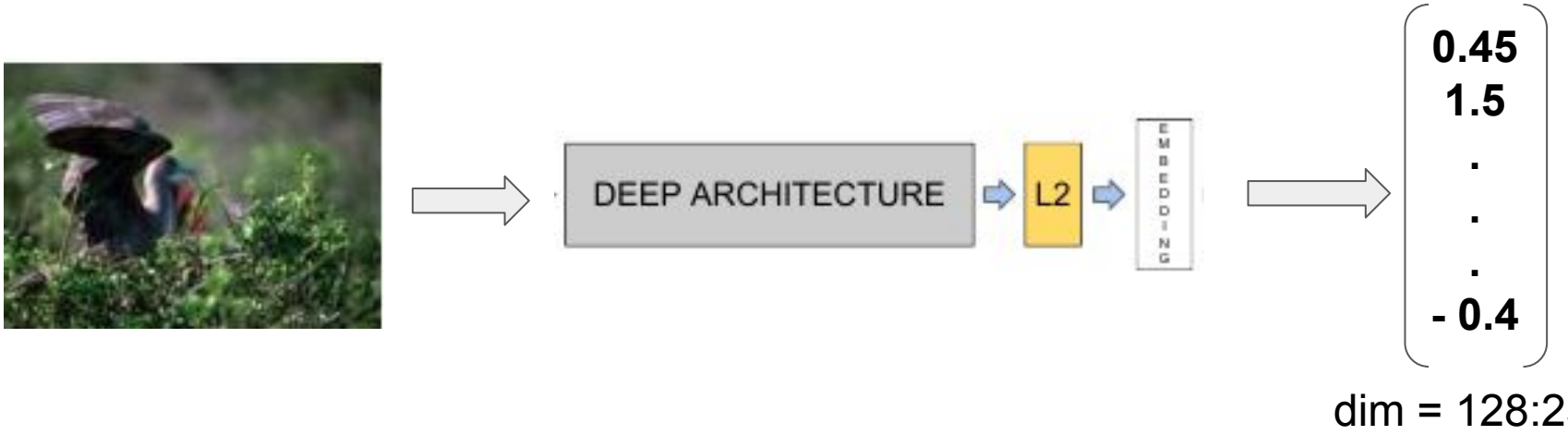
recall@K averages this score over all the images.

Metric Learning [Descriptor]

$$\begin{pmatrix} 0.45 \\ 1.5 \\ . \\ . \\ . \\ -0.4 \end{pmatrix}$$

dim = 128:256

Metric Learning [Descriptor]



dim = 128:256

In deep metric learning, feature embedding function **F** is modeled as a deep neural network. **F** embeds input images into feature embedding space with a certain desired condition. In this condition, the feature embeddings of similar images are required to be close to each other while those of dissimilar images are required to be far from each other.

Metric Learning [Contrastive Loss]

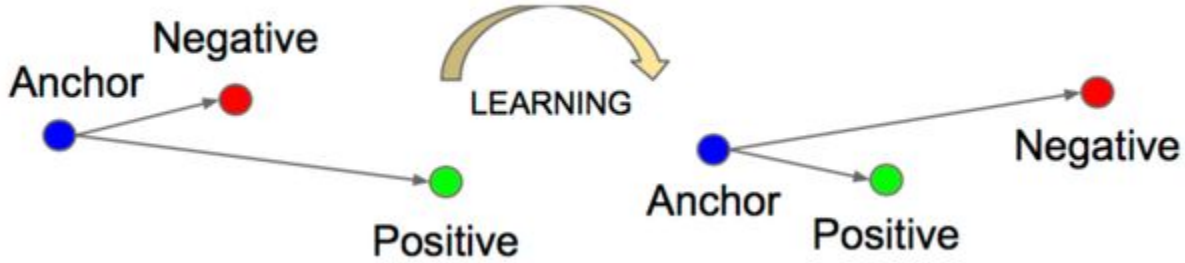
$$J = \frac{1}{m} \sum_{i,j} y_{i,j} D_{i,j}^2 + (1 - y_{i,j})[\alpha - D_{i,j}]_+^2$$

$$D_{i,j} = \|F(x_i) - F(x_j)\|_2$$

$$y_{i,j} = \mathbb{I}[y_i == y_j]$$

where *m* is a number of elements in batch,
α margin parameter

Metric Learning [Contrastive Loss]



$$J = \frac{3}{m} \sum_{i=1}^{m/3} [D_{a_i,p_i} - D_{a_i,n_i} + \alpha]_+$$

$$D_{a_i,p_i} = \left\| f(x_i^a) - f(x_i^p) \right\|_2 , y_i^a = y_i^p$$

$$D_{a_i,n_i} = \left\| f(x_i^a) - f(x_i^n) \right\|_2 , y_i^a \neq y_i^p$$

Metric Learning [Results]

Table 6. Recall@K(%) score on Stanford online products dataset (SOP)

K	1	10	100	1000
Contrastive^{128} [6]	42.0	58.2	73.8	89.1
LiftedStruct^{512} [6]	62.1	79.8	91.3	97.4
N-Pairs^{512} [8]	67.7	83.8	93.0	97.8
Clustering^{64} [9]	67.0	83.7	93.2	-
Proxy NCA†^{64} [36]	73.7	-	-	-
Margin†^{128} [33]	72.7	86.2	93.8	98.0
HDC^{384} [14]	69.5	84.4	92.8	97.7
A-Bier^{512} [39]	74.2	86.9	94.0	97.8
ABE-2^{512}	75.4	88.0	94.7	**98.2**
ABE-4^{512}	<u>75.9</u>	<u>88.3</u>	<u>94.8</u>	**98.2**
ABE-8^{512}	**76.3**	**88.4**	**94.8**	**98.2**

arXiv:1804.00382

Contents

Relational Reasoning

"Man sitting on bench"

objects

+ attributes

+ relationships

Justin Johnson, ICCV 2017, tutorial

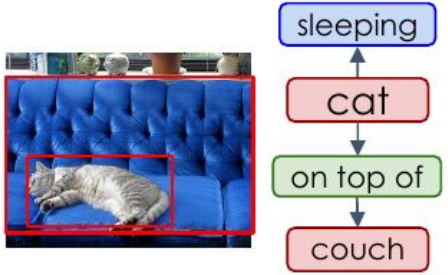
Relational Reasoning

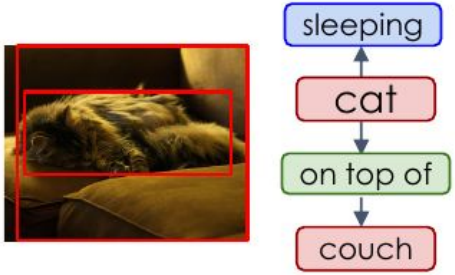
"Sleeping cat on top of couch"

objects

+ attributes

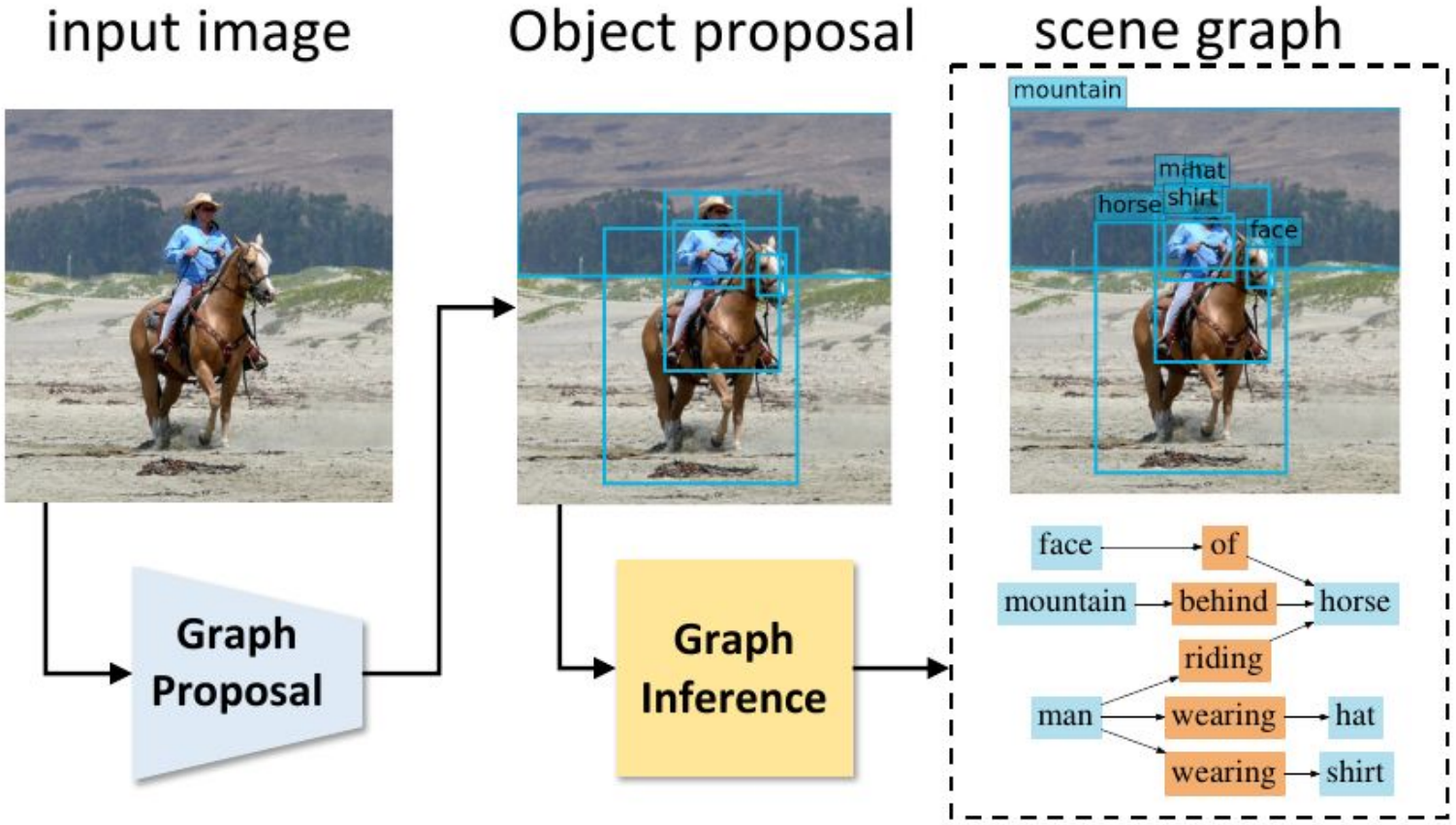
+ relationships





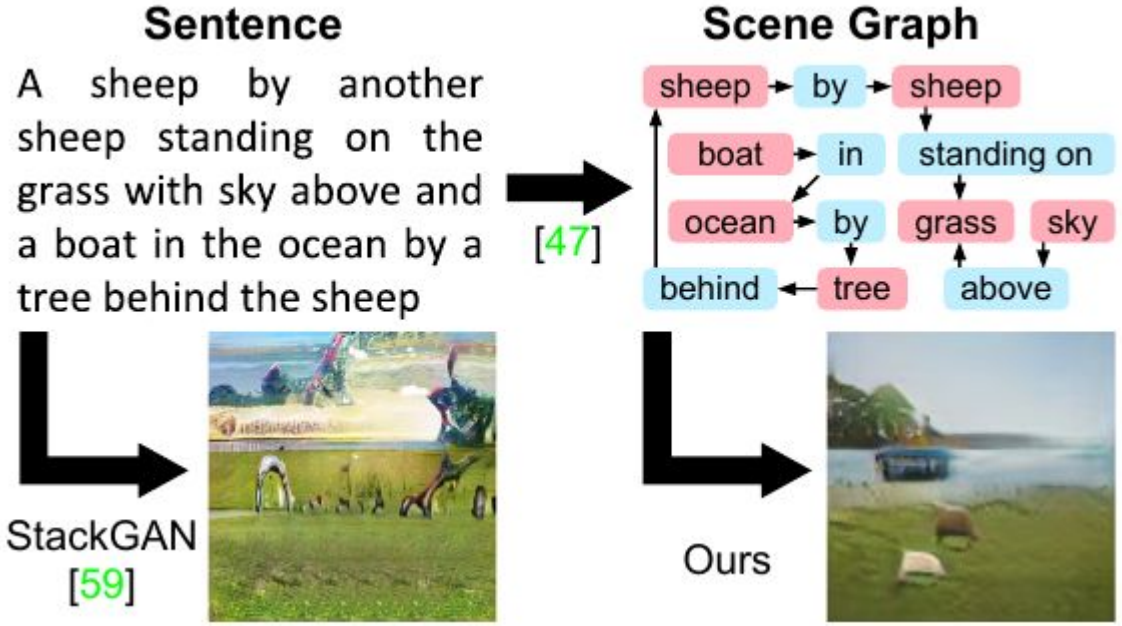
Justin Johnson, ICCV 2017, tutorial

Scene Understanding



Justin Johnson, ICCV 2017, tutorial

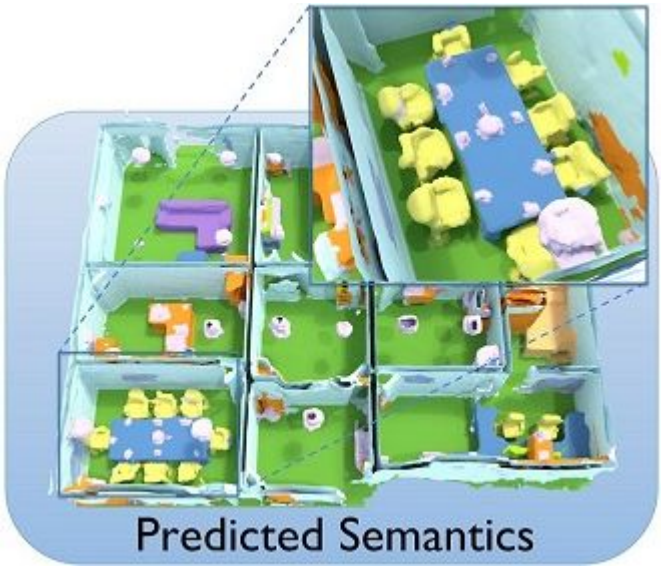
Scene Generation



arXiv: 1804.01622

3D Machine Learning

Car

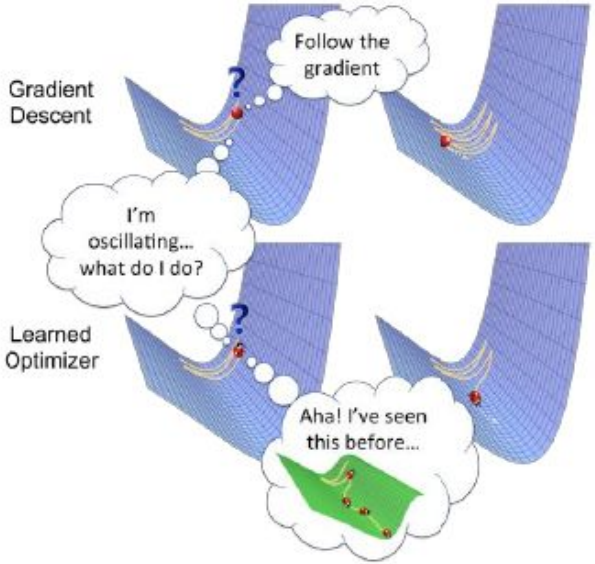


https://github.com/timzhang642/3D-Machine-Learning

Fei-Fei Li & Andrej Karpathy & Justin Johnson "CNN for Computer Vision" [cs231n]

Meta-learning

Meta learning is learning to learn.



- Optimizer Meta-Learning

- Few Shots Meta-Learning

Resources

- cs231n: Convolutional Neural Networks for Visual Recognition, http://cs231n.stanford.edu/
- Book “Deep Learning”, http://www.deeplearningbook.org/
- Arxiv Sanity Preserver, http://www.arxiv-sanity.com/top
- Google Scholar alerts
- Medium, blogs
- Twitter, Reddit, Quora