

# Outline

## 1 Foundations of CNN

- Convolution layer
- Pooling layer
- Data preprocessing

## 2 Famous CNN

- LeNet (1998)
- AlexNet (2012)
- ZFNet (2013)
- VGGNet (2014)
- GoogLeNet (2014)
- ResNet (2016)
- DenseNet (2017)
- Many other CNN

## 3 Applications

- Image classification
- Pose, action detection
- Object detection
- Scene labeling - Semantic segmentation
- Object tracking - videos
- Text detection and recognition

# Applications

This section is based on ["Recent advances in convolutional neural networks", Gu et al. 2015].

More applications domain and more references are presented in this paper.



# Outline

- 1 Foundations of CNN
  - Convolution layer
  - Pooling layer
  - Data preprocessing

- 2 Famous CNN
  - LeNet (1998)
  - AlexNet (2012)
  - ZFNet (2013)
  - VGGNet (2014)
  - GoogLeNet (2014)
  - ResNet (2016)
  - DenseNet (2017)
  - Many other CNN

- 3 Applications
  - **Image classification**
  - Pose, action detection
  - Object detection
  - Scene labeling - Semantic segmentation
  - Object tracking - videos
  - Text detection and recognition

## Image classification - Hierarchy of classifiers

[“Error-driven incremental learning in deep convolutional neural network for large-scale image classification”, Xiao et al. 2014]

→ They propose a training method that grows a network not only incrementally but also hierarchically. In their method, **classes are grouped according to similarities** and are self-organized into different levels.

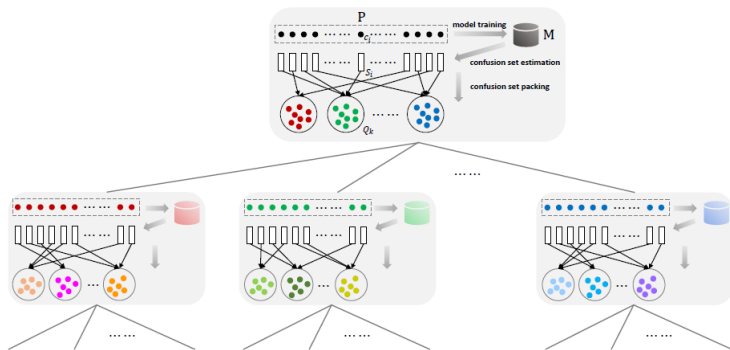
[“HD-CNN: hierarchical deep convolutional neural networks for large scale visual recognition”, Yan et al. 2015]

→ They introduce a hierarchical deep CNNs (HD-CNNs) by embedding deep CNNs into a category hierarchy. They decompose the classification task into two steps. The coarse category CNN classifier is first used to **separate easy classes from each other**, and then those more **challenging classes are routed downstream to fine category classifiers** for further prediction. This architecture follows the coarse-to-fine classification paradigm and can achieve lower error at the cost of an affordable increase of complexity.



# Image classification - CNN Tree

Z. Wang et al. 2018 build a tree of CNN to learn fine-grained features for subcategory recognition.



# Image classification - CNN Tree

Category	Confusion Set									
 <p>loch</p>	 <p>gar</p>	 <p>sturgeon</p>	 <p>coho</p>	 <p>sal</p>	 <p>barracouta</p>					
 <p>indigo bunting</p>	 <p>European gull-nile</p>	 <p>jacamar</p>	 <p>peacock</p>	 <p>chual</p>	 <p>macaw</p>	 <p>jay</p>				
 <p>red-breasted mergan</p>	 <p>albatross</p>	 <p>peikan</p>	 <p>oystercatcher</p>	 <p>drake</p>	 <p>radshank</p>	 <p>goose</p>	 <p>American coot</p>			
 <p>echidna</p>	 <p>porcupine</p>	 <p>beaver</p>	 <p>armadillo</p>	 <p>mongoose</p>						
 <p>shopping basket</p>	 <p>bucket</p>	 <p>shopping cart</p>	 <p>packet</p>	 <p>mail bag</p>	 <p>hamper</p>	 <p>grocery store</p>				

Figure: Confusion set outputs by AlexNet softmax prediction on validation set of ILSVRC 2015.

# Image classification - CNN Tree

		AlexNet					
	#Basic Models	1	2	3	4	5	6
Top-1 errors	$T_0$	43.09%	41.28%	40.41%	40.21%	39.82%	39.63%
	$T_1$	40.68%(-2.41%)	38.95%(-2.33%)	38.07%(-2.34%)	37.80%(-2.41%)	37.49%(-2.33%)	37.39%(-2.24%)
	$T_2$	40.40%(-2.69%)	38.60%(-2.68%)	37.84%(-2.57%)	37.62%(-2.59%)	37.33%(-2.49%)	37.19%(-2.44%)
Top-5 errors	$T_0$	20.04%	18.53%	18.03%	17.72%	17.52%	17.43%
	$T_1$	18.58%(-1.46%)	17.52%(-1.01%)	16.93%(-1.10%)	16.59%(-1.13%)	16.39%(-1.13%)	16.24%(-1.19%)
	$T_2$	18.55%(-1.49%)	17.39%(-1.14%)	16.81%(-1.22%)	16.53%(-1.19%)	16.36%(-1.16%)	16.23%(-1.20%)

		GoogleNet					
	#Basic Models	1	2	3	4	5	6
Top-1 errors	$T_0$	32.75%	30.96%	30.27%	29.89%	29.72%	29.56%
	$T_1$	28.37%(-4.38%)	26.51%(-4.45%)	25.99%(-4.28%)	25.57%(-4.32%)	25.4%(-4.32%)	25.15%(-4.41%)
Top-5 errors	$T_0$	12.00%	10.89	10.53%	10.32%	10.17%	10.08%
	$T_1$	10.09%(-1.91%)	8.98%(-1.91%)	8.68%(-1.85%)	8.33%(-1.99%)	8.23%(-1.94%)	8.12%(-1.96%)

# Image classification - CNN Tree






















Category	Example Validation Images					
barracouta						
church						
spaghetti squash						
espresso						
trolleybus						

Figure: Top label is given by basic AlexNet CNN while bottom one is given by CNNTree (green color corresponds to a correct prediction)

# Outline

## 1 Foundations of CNN

- Convolution layer
- Pooling layer
- Data preprocessing

## 2 Famous CNN

- LeNet (1998)
- AlexNet (2012)
- ZFNet (2013)
- VGGNet (2014)
- GoogLeNet (2014)
- ResNet (2016)
- DenseNet (2017)
- Many other CNN

## 3 Applications

- Image classification
- **Pose, action detection**
- Object detection
- Scene labeling - Semantic segmentation
- Object tracking - videos
- Text detection and recognition

# Pose estimation - Deeppose

["Deeppose: Human pose estimation via deep neural networks", Toshev and Szegedy 2014]

DeepPose is the first application of CNNs to human pose estimation problem. It captures the full context of each body joint by taking the whole image as the input.

Previous works:

- Limited expressiveness – the use of local detectors, which reason in many cases about a single part
- Modeling only a small subset of all interactions between body parts.

# Pose estimation - Deeppose

Structure:

- Normalizing images
- Regression problem, i.e., prediction of  $k$  joints

$$\text{Image} \mapsto \mathbf{y} \in \mathbb{R}^{2k}.$$

- Use a cascade of 7 layers, each one taking a zoom of the previous image as input (refinement of the prediction at each stage).



## Pose estimation - Deeppose

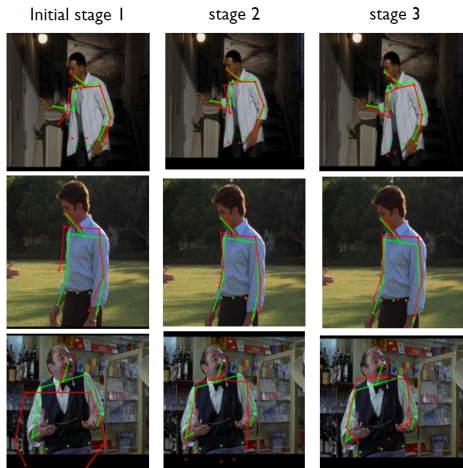


Figure 6. Predicted poses in red and ground truth poses in green for the first three stages of a cascade for three examples.



# Pose estimation - Deeppose



## Action recognition - images

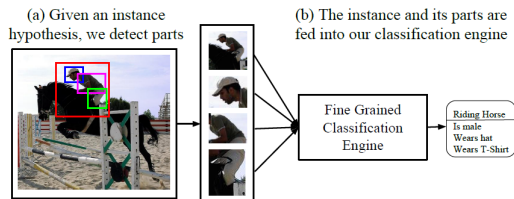
**Action recognition** aims at classifying human activities based on their visual appearance and motion dynamics.

In Simonyan and Zisserman 2014b (VGG), they use the outputs of the penultimate layer of a pre-trained CNN to predict actions and achieve a high level of performance in action classification.

Gkioxari et al. 2015 add a part detection to this framework. Their part detector is a CNN based extension to the original Poselet Pishchulin et al. 2013 method.

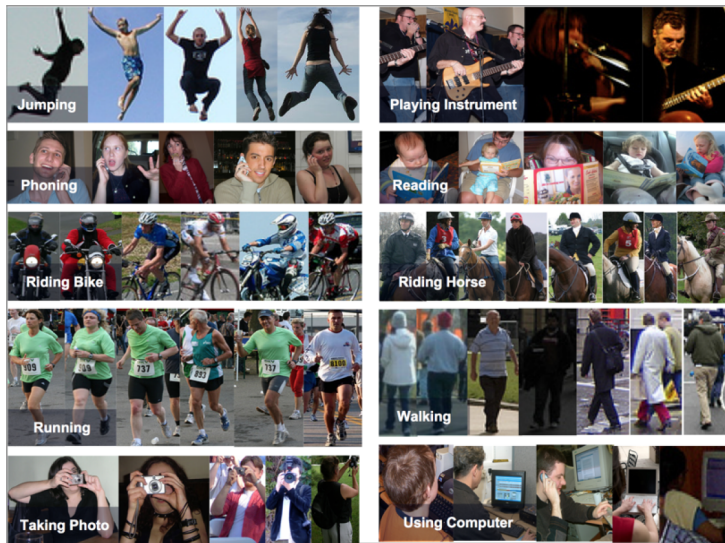
# Action recognition

[“Actions and attributes from wholes and parts”, Gkioxari et al. 2015]



Given an R-CNN person detection (red box), they detect parts using a novel, deep version of poselets. The detected whole-person and part bounding boxes are input into a fine-grained classification engine to produce predictions for actions and attributes.

# Action recognition



# Outline

## 1 Foundations of CNN

- Convolution layer
- Pooling layer
- Data preprocessing

## 2 Famous CNN

- LeNet (1998)
- AlexNet (2012)
- ZFNet (2013)
- VGGNet (2014)
- GoogLeNet (2014)
- ResNet (2016)
- DenseNet (2017)
- Many other CNN

## 3 Applications

- Image classification
- Pose, action detection
- **Object detection**
- Scene labeling - Semantic segmentation
- Object tracking - videos
- Text detection and recognition

## Object detection - Exhaustive search vs segmentation

Segmentation: aims for a unique partitioning of the image through a generic algorithm, where there is one part for all object silhouettes in the image.



(a)

(b)



(c)

(d)

High variety of reasons that an image region forms an object:

[“Selective search for object recognition”, Uijlings et al. 2013]

## Object detection - Exhaustive search vs segmentation

Segmentation: aims for a unique partitioning of the image through a generic algorithm, where there is one part for all object silhouettes in the image.



(a)

(b)



(c)

(d)

High variety of reasons that an image region forms an object:

- (b) the cats can be distinguished by colour, not texture.
- (c) the chameleon can be distinguished from the surrounding leaves by texture, not colour.
- (d) the wheels can be part of the car because they are enclosed, not because they are similar in texture or colour.
- (a) many different scales needed

[“Selective search for object recognition”, Uijlings et al. 2013]

→ Necessity to use a variety of diverse strategies.

## Object detection - Exhaustive search vs segmentation

**Alternative approach:** do localisation through the identification of an object.

**Exhaustive search:** With an appearance model learned from examples, an exhaustive search is performed where every location within the image is examined as to not miss any potential object location.

Searching every possible location is computationally infeasible.

→ restrictions need to be imposed: the classifier is simplified and the appearance model needs to be fast.

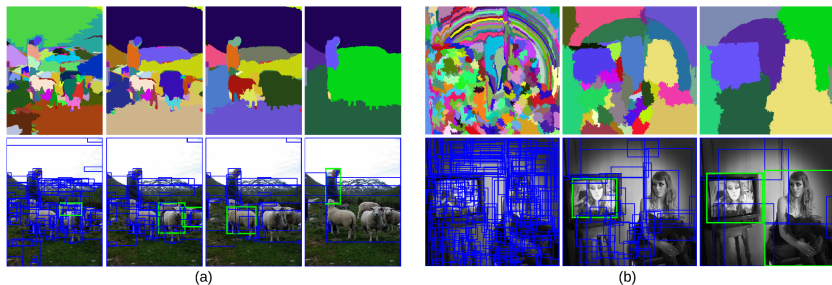
**Selective search:** data-driven selective search using bottom up grouping.



# Object detection - Exhaustive search vs segmentation

Bottom-up grouping generates hierarchical nested partitioning of the input image.

["Mean shift: A robust approach toward feature space analysis"; "Efficient graph-based image segmentation", Comaniciu and Meer 2002; Felzenszwalb and Huttenlocher 2004]



# Object detection - Exhaustive search vs segmentation

## Generic algorithm:

- They first use Felzenszwalb and Huttenlocher 2004 to create initial regions. This method is the fastest, publicly available algorithm that yields high quality starting locations.
- Then they use a greedy algorithm to iteratively group regions together
  - ▶ First the similarities between all neighbouring regions are calculated.
  - ▶ The two most similar regions are grouped together, and new similarities are calculated between the resulting region and its neighbours.
  - ▶ The process of grouping the most similar regions is repeated until the whole image becomes a single region.

## Variety of partitionings:

- Different variant of input images
- Similarities based on color, texture, size, shared pixels

<i>colour spaces</i>	RGB	I	Lab	rgI	HSV	rgb	C	H
Light Intensity	-	-	+/-	2/3	2/3	+	+	+
Shadows/shading	-	-	+/-	2/3	2/3	+	+	+
Highlights	-	-	-	-	1/3	-	+/-	+

## Object detection - naive approach

Generally, the difficulties mainly lie in how to accurately and efficiently localize objects in images or video frames.

In some early works by Vaillant et al. 1994; Nowlan and Platt 1995; Girshick, Iandola, et al. 2015, they use the sliding window based approaches to densely evaluate the CNN classifier on windows sampled at each location and scale. Since there are usually hundreds of thousands of candidate windows in a image, these methods suffer from highly computational cost, which makes them unsuitable to be applied on the large-scale dataset

More references on object proposal based methods:

["Human detection from images and videos: A survey", Nguyen et al. 2016]

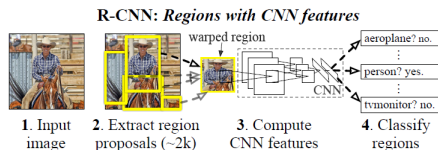
["Category-independent object proposals with diverse ranking", Endres and Hoiem 2014]

["Textproposals: a text-specific selective search algorithm for word spotting in the wild", Gómez and Karatzas 2017]

# Object detection - R-CNN - Regions with CNN features

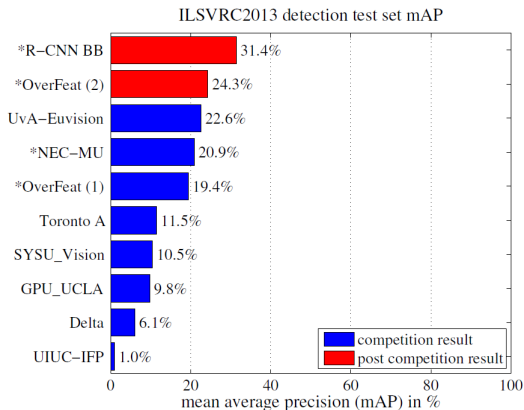
One of the most famous object proposal based CNN detector is Region-based CNN (R-CNN) by Girshick, Donahue, et al. 2014, aiming at

- localizing objects with a deep network
- training a high-capacity model with only a small quantity of annotated detection data



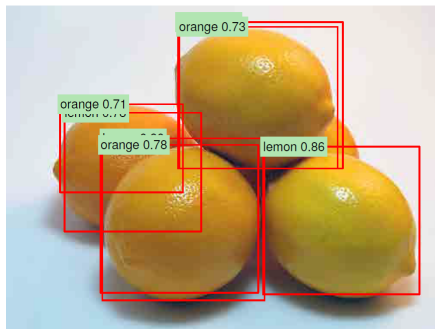
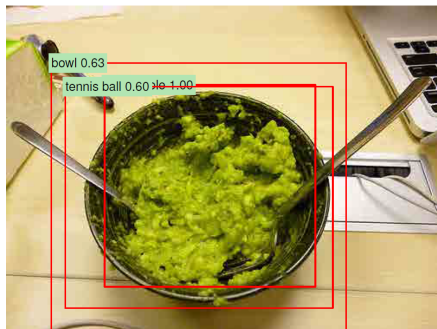
1. Generating category-independent region proposals via selective search.
2. Training large CNN that extracts a fixed-length feature vector from each region (Supervised pre-training on the large auxiliary dataset ILSVRC, followed by domainspecific fine-tuning on the small dataset PASCAL).
3. Learning a set of class-specific linear SVMs.

# Object detection - R-CNN - Regions with CNN features

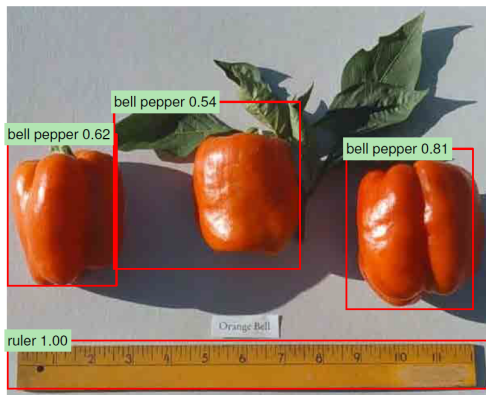


However, computational cost is high since the time-consuming CNN feature extractor will be performed for each region separately.

# Object detection - R-CNN - Regions with CNN features



# Object detection - R-CNN - Regions with CNN features



# Object detection - improving R-CNN

[“Spatial pyramid pooling in deep convolutional networks for visual recognition”, He et al. 2014]

**Spatial Pyramid Pooling** network (SPP net) is a pyramid-based version of R-CNN, which introduces an SPP layer to relax the constraint that input images must have a fixed size. Unlike R-CNN, SPP net extracts the feature maps from the entire image only once, and then applies spatial pyramid pooling on each candidate window to get a fixed-length representation.

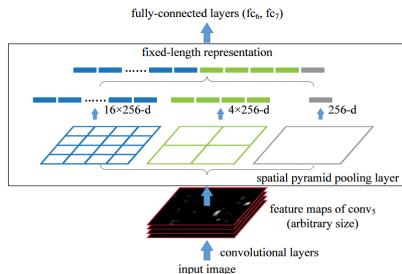


Figure 3: A network structure with a **spatial pyramid pooling layer**. Here 256 is the filter number of the conv<sub>5</sub> layer, and conv<sub>5</sub> is the last convolutional layer.



## Object detection - improving R-CNN

**Drawback:** multi-stage pipeline  $\Rightarrow$  CNN feature extractor and SVM classifier are impossible to train jointly.

[“Faster r-cnn: Towards real-time object detection with region proposal networks”, Ren et al. 2015]

**Faster RCNN** improves SPP net by using an end-to-end training method. All network layers can be updated during fine-tuning, which simplifies the learning process and improves detection accuracy.

[“Attentionnet: Aggregating weak directions for accurate object detection”, Yoo et al. 2015]

They treat the object detection problem as an iterative classification problem. It predicts an accurate object boundary box by aggregating quantized weak directions from their detection network.

## Object detection - YOLO, SSD

More recently, YOLO Redmon et al. 2016 and SSD W. Liu et al. 2016 allow single pipeline detection that directly predicts class labels.

**YOLO (You Only Look Once)** treats object detection as a regression problem to spatially separated bounding boxes and associated class probabilities.

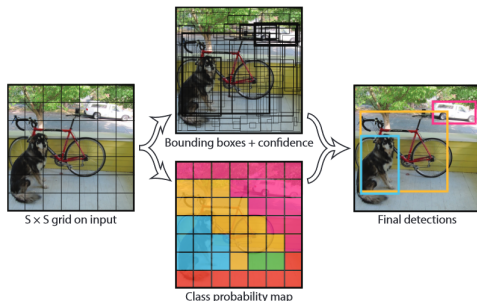
**SSD (Single Shot Detector)** discretizes the output space of bounding boxes into a set of default boxes over different aspect ratios and scales per feature map location. With this multiple scales setting and their matching strategy, SSD is significantly more accurate than YOLO.

With the benefits from super-resolution, Lu et al. 2016 propose a top-down search strategy to divide a window into sub-windows recursively, in which an additional network is trained to account for such division decisions.

# YOLO

["You only look once: Unified, real-time object detection", Redmon et al. 2016]

The whole detection pipeline is a single network which predicts bounding boxes and class probabilities from the full image in one evaluation, and can be optimized end-to-end directly on detection performance.



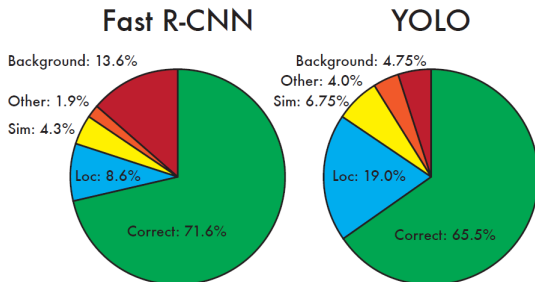
## Drawback

Fails to detect small numerous objects.

**Figure 2: The Model.** Our system models detection as a regression problem. It divides the image into an  $S \times S$  grid and for each grid cell predicts  $B$  bounding boxes, confidence for those boxes, and  $C$  class probabilities. These predictions are encoded as an  $S \times S \times (B * 5 + C)$  tensor.

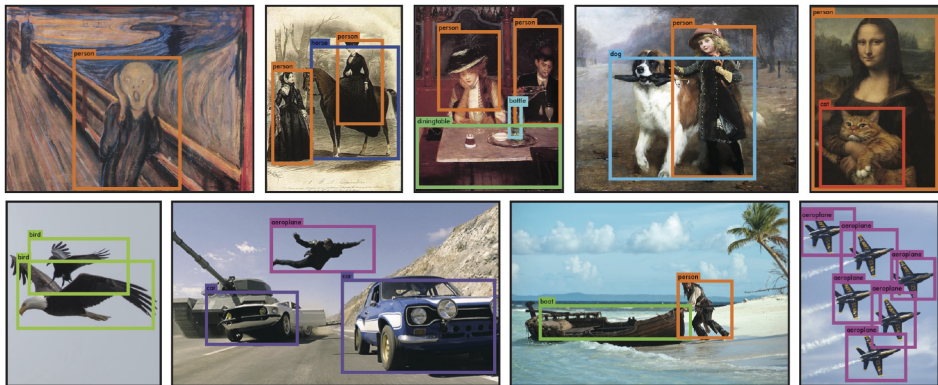
# YOLO

YOLO still lags behind state-of-the-art detection systems in accuracy. While it can quickly identify objects in images it struggles to precisely localize some objects, especially small ones.

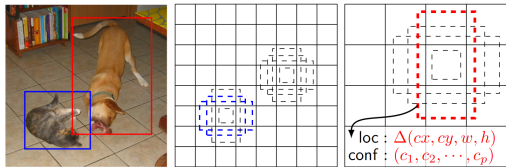
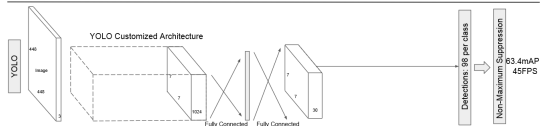
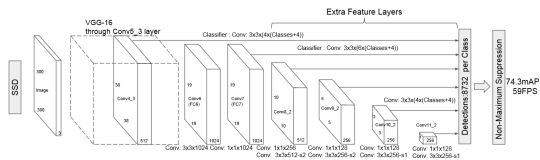


**Figure 4: Error Analysis: Fast R-CNN vs. YOLO** These charts show the percentage of localization and background errors in the top N detections for various categories ( $N = \#$  objects in that category).

# YOLO

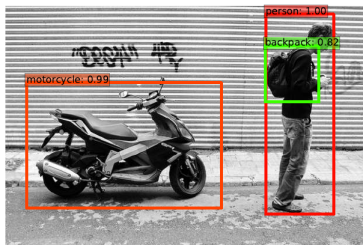
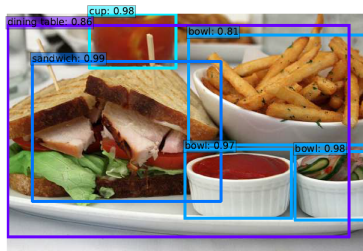
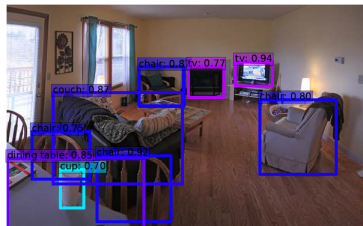
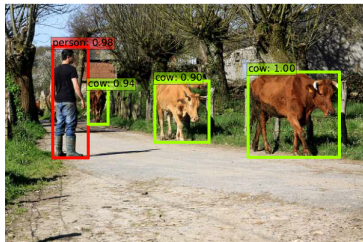


[“Ssd: Single shot multibox detector”, W. Liu et al. 2016]



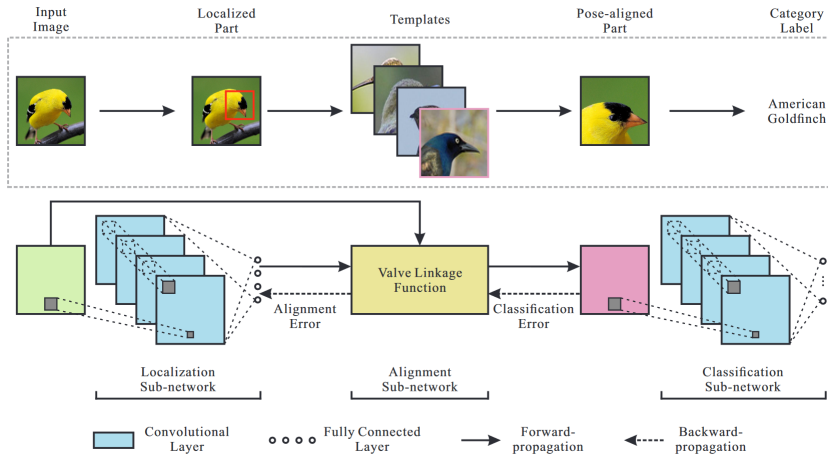
(a) Image with GT boxes (b)  $8 \times 8$  feature map (c)  $4 \times 4$  feature map

- SSD is also a single shot detector (i.e. with no region proposals) contrary to R-CNN.
- SSD uses convolutional layers at the end of the network (contrary to YOLO that uses fully connected layers)
- In SSD, the end of the network is composed of feature maps of different sizes. Using these feature maps allows to capture objects of different sizes, contrary to YOLO which uses one single grid on the input image.



# Image classification - Going further

Lin et al. 2015 incorporate part localization, alignment, and classification into one recognition system which is called Deep LAC.





## Image classification - Going further

Annotations are not easy to collect and these systems have difficulty in scaling up and to handle many types of fine-grained classes.

["Fine-grained recognition without part annotations", Krause et al. 2015] combine co-segmentation and alignment in a discriminative mixture to generate parts for facilitating fine-grained classification.

["Weakly supervised fine-grained categorization with part-based image representation", Zhang et al. 2016] use the unsupervised selective search to generate object proposals, and then select the useful parts from the multi-scale generated part proposals.

Object detection and classification: see also ["Deep neural networks for object detection", Szegedy, Toshev, et al. 2013]

# Outline

## 1 Foundations of CNN

- Convolution layer
- Pooling layer
- Data preprocessing

## 2 Famous CNN

- LeNet (1998)
- AlexNet (2012)
- ZFNet (2013)
- VGGNet (2014)
- GoogLeNet (2014)
- ResNet (2016)
- DenseNet (2017)
- Many other CNN

## 3 Applications

- Image classification
- Pose, action detection
- Object detection
- **Scene labeling - Semantic segmentation**
- Object tracking - videos
- Text detection and recognition

# Scene labeling

**Scene labeling** aims to relate one semantic class (road, water, sea...) to each pixel of the input image

→ ["Recurrent convolutional neural networks for scene labeling", Pinheiro and Collobert 2014]

To enable the CNNs to have a large field of view over pixels, they develop the recurrent CNNs. More specifically, the identical CNNs are applied recurrently to the output maps of CNNs in the previous iterations. By doing this, they can achieve slightly better labelling results while significantly reducing the inference times.

→ ["Dag-recurrent neural networks for scene labeling", Shuai et al. 2016]

They use the recurrent neural networks to model the contextual dependencies among image features from CNNs, and dramatically boost the labelling performance.

## Object semantic segmentation

["Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs", L.-C. Chen et al. 2018]

They apply pre-trained deep CNNs to emit the labels of pixels. Considering that the imperfectness of boundary alignment, they further use fully connected Conditional Random Field (CRF) to boost the labelling performance.

# Scene labeling - DAG-RNN

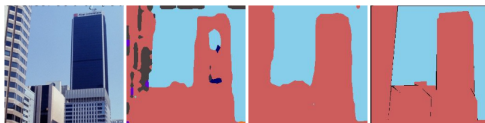


Input Image

CNN

DAG-RNN

Ground Truth



Input Image

CNN

DAG-RNN

Ground Truth

# Outline

- 1 Foundations of CNN
  - Convolution layer
  - Pooling layer
  - Data preprocessing

- 2 Famous CNN
  - LeNet (1998)
  - AlexNet (2012)
  - ZFNet (2013)
  - VGGNet (2014)
  - GoogLeNet (2014)
  - ResNet (2016)
  - DenseNet (2017)
  - Many other CNN

- 3 Applications
  - Image classification
  - Pose, action detection
  - Object detection
  - Scene labeling - Semantic segmentation
  - **Object tracking - videos**
  - Text detection and recognition

# Object tracking

The success in object tracking relies heavily on how robust the representation of target appearance is against several challenges such as view point changes, illumination changes, and occlusions

[“Deeptrack: Learning discriminative feature representations online for robust visual tracking”, Li et al. 2016]

They propose a target-specific CNN for object tracking, where the CNN is trained incrementally during tracking with new examples obtained online. They employ a candidate pool of multiple CNNs as a data-driven model of different instances of the target object.

[“Cnntracker: Online discriminative object tracking via deep convolutional neural network”, Y. Chen et al. 2016]

A CNN object tracking method is proposed to address limitations of handcrafted features and shallow classifier structures in object tracking problem.

[“Online tracking by learning discriminative saliency map with convolutional neural network”, Hong et al. 2015]

They propose a visual tracking algorithm based on a pre-trained CNN. They put an additional layer of an online SVM to learn a target appearance discriminatively against background.

<https://pjreddie.com/darknet/yolo/>

## Pose/Action recognition - videos

Applying CNNs on videos is challenging because traditional CNNs are designed to represent two-dimensional spatial signals but in videos a new temporal axis is added which is essentially different from a spatial dimension.

[“3D convolutional neural networks for human action recognition”, Ji et al. 2013]

They consider the temporal axis in a similar manner as other spatial axes and introduce a network of 3D convolutional layers to be applied on video inputs.

[“Two-stream convolutional networks for action recognition in videos”, Simonyan and Zisserman 2014a]

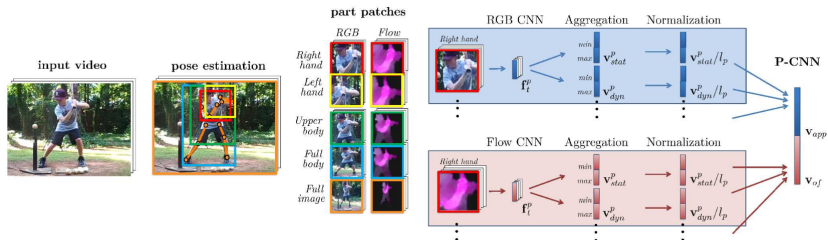
Separating the representation to spatial and temporal variations and train individual CNNs for each of them. First stream of this framework is a traditional CNN applied on all the frames and the second receives the dense optical flow of the input videos and trains another CNN which is identical to the spatial stream in size and structure. The output of the two streams are combined in a class score fusion step.

[“P-cnn: Pose-based cnn features for action recognition”, Chéron et al. 2015]

They use the two stream CNN on the localized parts of the human body and show the aggregation of part-based local CNN descriptors can effectively improve the performance of action recognition.

# Pose estimation - P-CNN

[“P-cnn: Pose-based cnn features for action recognition”, Chéron et al. 2015]



[“End-to-end learning of deformable mixture of parts and deep convolutional neural networks for human pose estimation”, W. Yang et al. 2016]

<https://www.youtube.com/watch?v=MKVvQK8FawE>

[“Segnet: A deep convolutional encoder-decoder architecture for image segmentation”, Badrinarayanan et al. 2015]

[https://www.youtube.com/watch?v=CxanE\\_W46ts](https://www.youtube.com/watch?v=CxanE_W46ts)

[“Realtime multi-person 2d pose estimation using part affinity fields”, Cao et al. 2016]

<https://www.youtube.com/watch?v=pW6nZXeWlGM>



# Outline

## 1 Foundations of CNN

- Convolution layer
- Pooling layer
- Data preprocessing

## 2 Famous CNN

- LeNet (1998)
- AlexNet (2012)
- ZFNet (2013)
- VGGNet (2014)
- GoogLeNet (2014)
- ResNet (2016)
- DenseNet (2017)
- Many other CNN

## 3 Applications

- Image classification
- Pose, action detection
- Object detection
- Scene labeling - Semantic segmentation
- Object tracking - videos
- Text detection and recognition

# Text detection and recognition

Optical Character Recognition (OCR) can be categorized into three types:

- 1 text detection and localization without recognition,
- 2 text recognition on cropped text images,
- 3 end-to-end text spotting that integrates both text detection and recognition.

Several proposed methods:

- CNN model originally trained for character classification to perform text detection  
[“End-to-end text recognition with convolutional neural networks”, T. Wang et al. 2012]
- CNN model allowing feature sharing across four different subtask: text detection, character case-sensitive and insensitive classification, and bigram classification.  
[“Deep features for text spotting”, Jaderberg, Vedaldi, et al. 2014]
- Elementary subtasks as text bounding box filtering, text bounding box regression, and text recognition are each tackled by a separate CNN model.  
[“Reading text in the wild with convolutional neural networks”, Jaderberg, Simonyan, et al. 2016]

## References

- [Bis95] Chris M Bishop. “Training with noise is equivalent to Tikhonov regularization”. In: *Neural computation* 7.1 (1995), pp. 108–116.
- [BKC15] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. “Segnet: A deep convolutional encoder-decoder architecture for image segmentation”. In: *arXiv preprint arXiv:1511.00561* (2015).
- [Bre00] Leo Breiman. “Randomizing outputs to increase prediction accuracy”. In: *Machine Learning* 40.3 (2000), pp. 229–242.
- [Cao+16] Zhe Cao et al. “Realtime multi-person 2d pose estimation using part affinity fields”. In: *arXiv preprint arXiv:1611.08050* (2016).
- [Che+16] Yan Chen et al. “Cnntracker: Online discriminative object tracking via deep convolutional neural network”. In: *Applied Soft Computing* 38 (2016), pp. 1088–1098.
- [Che+18] Liang-Chieh Chen et al. “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs”. In: *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2018), pp. 834–848.

## References

- [Cho17] François Chollet. “Xception: Deep learning with depthwise separable convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.
- [CLS15] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid. “P-cnn: Pose-based cnn features for action recognition”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 3218–3226.
- [CM02] Dorin Comaniciu and Peter Meer. “Mean shift: A robust approach toward feature space analysis”. In: *IEEE Transactions on pattern analysis and machine intelligence* 24.5 (2002), pp. 603–619.
- [CPC16] Alfredo Canziani, Adam Paszke, and Eugenio Culurciello. “An analysis of deep neural network models for practical applications”. In: *arXiv preprint arXiv:1605.07678* (2016).
- [EF15] David Eigen and Rob Fergus. “Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2650–2658.

## References

- [EH14] Ian Endres and Derek Hoiem. “Category-independent object proposals with diverse ranking”. In: *IEEE transactions on pattern analysis and machine intelligence* 36.2 (2014), pp. 222–234.
- [FH04] Pedro F Felzenszwalb and Daniel P Huttenlocher. “Efficient graph-based image segmentation”. In: *International journal of computer vision* 59.2 (2004), pp. 167–181.
- [GGM15] Georgia Gkioxari, Ross Girshick, and Jitendra Malik. “Actions and attributes from wholes and parts”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2470–2478.
- [Gir+14] Ross Girshick, Jeff Donahue, et al. “Rich feature hierarchies for accurate object detection and semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 580–587.
- [Gir+15] Ross Girshick, Forrest Iandola, et al. “Deformable part models are convolutional neural networks”. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2015, pp. 437–446.
- [GK17] Lluís Gómez and Dimosthenis Karatzas. “Textproposals: a text-specific selective search algorithm for word spotting in the wild”. In: *Pattern Recognition* 70 (2017), pp. 60–74.

## References

- [Gra11] Alex Graves. “Practical variational inference for neural networks”. In: *Advances in Neural Information Processing Systems*. 2011, pp. 2348–2356.
- [GSS14] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. “Explaining and harnessing adversarial examples”. In: *arXiv preprint arXiv:1412.6572* (2014).
- [Gu+15] Jiuxiang Gu et al. “Recent advances in convolutional neural networks”. In: *arXiv preprint arXiv:1512.07108* (2015).
- [He+14] Kaiming He et al. “Spatial pyramid pooling in deep convolutional networks for visual recognition”. In: *European conference on computer vision*. Springer. 2014, pp. 346–361.
- [He+15] Kaiming He et al. “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034.
- [He+16] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.
- [Hon+15] Seunghoon Hong et al. “Online tracking by learning discriminative saliency map with convolutional neural network”. In: *International Conference on Machine Learning*. 2015, pp. 597–606.

## References

- [HOT06] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. “A fast learning algorithm for deep belief nets”. In: *Neural computation* 18.7 (2006), pp. 1527–1554.
- [Hua+17] Gao Huang et al. “Densely Connected Convolutional Networks.”. In: *CVPR*. Vol. 1. 2. 2017, p. 3.
- [Jad+16] Max Jaderberg, Karen Simonyan, et al. “Reading text in the wild with convolutional neural networks”. In: *International Journal of Computer Vision* 116.1 (2016), pp. 1–20.
- [JGH96] Kam-Chuen Jim, C Lee Giles, and Bill G Horne. “An analysis of noise in recurrent neural networks: convergence and generalization”. In: *IEEE Transactions on neural networks* 7.6 (1996), pp. 1424–1438.
- [Ji+13] Shuiwang Ji et al. “3D convolutional neural networks for human action recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.1 (2013), pp. 221–231.
- [JKL+09] Kevin Jarrett, Koray Kavukcuoglu, Yann LeCun, et al. “What is the best multi-stage architecture for object recognition?” In: *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE. 2009, pp. 2146–2153.

## References

- [JVZ14] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. “Deep features for text spotting”. In: *European conference on computer vision*. Springer. 2014, pp. 512–528.
- [Kha+20] Asifullah Khan et al. “A survey of the recent architectures of deep convolutional neural networks”. In: *Artificial Intelligence Review* 53.8 (2020), pp. 5455–5516.
- [Kra+15] Jonathan Krause et al. “Fine-grained recognition without part annotations”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 5546–5555.
- [KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.
- [LeC+89] Yann LeCun et al. “Generalization and network design strategies”. In: *Connectionism in perspective* (1989), pp. 143–155.
- [LeC+98] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.
- [Lee+15] Chen-Yu Lee et al. “Deeply-supervised nets”. In: *Artificial Intelligence and Statistics*. 2015, pp. 562–570.



## References

- [Lin+15] Di Lin et al. “Deep lac: Deep localization, alignment and classification for fine-grained recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1666–1674.
- [Liu+16] Wei Liu et al. “Ssd: Single shot multibox detector”. In: *European conference on computer vision*. Springer. 2016, pp. 21–37.
- [LJL16] Yongxi Lu, Tara Javidi, and Svetlana Lazebnik. “Adaptive object detection using adjacency and zoom prediction”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 2351–2359.
- [LLP16] Hanxi Li, Yi Li, and Fatih Porikli. “Deeptrack: Learning discriminative feature representations online for robust visual tracking”. In: *IEEE Transactions on Image Processing* 25.4 (2016), pp. 1834–1848.
- [Mar10] James Martens. “Deep learning via Hessian-free optimization.”. In: *ICML*. Vol. 27. 2010, pp. 735–742.
- [NLO16] Duc Thanh Nguyen, Wanqing Li, and Philip O Ogunbona. “Human detection from images and videos: A survey”. In: *Pattern Recognition* 51 (2016), pp. 148–175.

## References

- [NP95] Steven J Nowlan and John C Platt. “A convolutional neural network hand tracker”. In: *Advances in neural information processing systems* (1995), pp. 901–908.
- [Pau+14] Mattis Paulin et al. “Transformation pursuit for image classification”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 3646–3653.
- [PC14] Pedro HO Pinheiro and Ronan Collobert. “Recurrent convolutional neural networks for scene labeling”. In: *31st International Conference on Machine Learning (ICML)*. EPFL-CONF-199822. 2014.
- [Pis+13] Leonid Pishchulin et al. “Poselet conditioned pictorial structures”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 588–595.
- [Red+16] Joseph Redmon et al. “You only look once: Unified, real-time object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 779–788.
- [Ren+15] Shaoqing Ren et al. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems*. 2015, pp. 91–99.

## References

- [Rif+11] Salah Rifai et al. “Adding noise to the input of a model trained with a regularized objective”. In: *arXiv preprint arXiv:1104.3250* (2011).
- [Rip07] Brian D Ripley. *Pattern recognition and neural networks*. Cambridge university press, 2007.
- [Rus+15] Olga Russakovsky et al. “Imagenet large scale visual recognition challenge”. In: *International Journal of Computer Vision* 115.3 (2015), pp. 211–252.
- [SB17] Justin Salamon and Juan Pablo Bello. “Deep convolutional neural networks and data augmentation for environmental sound classification”. In: *IEEE Signal Processing Letters* 24.3 (2017), pp. 279–283.
- [SGS15a] Rupesh K Srivastava, Klaus Greff, and Jürgen Schmidhuber. “Training very deep networks”. In: *Advances in neural information processing systems*. 2015, pp. 2377–2385.
- [SGS15b] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. “Highway networks”. In: *arXiv preprint arXiv:1505.00387* (2015).
- [Shu+16] Bing Shuai et al. “Dag-recurrent neural networks for scene labeling”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 3620–3629.

## References

- [STE13] Christian Szegedy, Alexander Toshev, and Dumitru Erhan. “Deep neural networks for object detection”. In: *Advances in neural information processing systems*. 2013, pp. 2553–2561.
- [SZ14a] Karen Simonyan and Andrew Zisserman. “Two-stream convolutional networks for action recognition in videos”. In: *Advances in neural information processing systems*. 2014, pp. 568–576.
- [SZ14b] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).
- [Sze+15] Christian Szegedy, Wei Liu, et al. “Going deeper with convolutions”. In: *Cvpr*. 2015.
- [Sze+16] Christian Szegedy, Vincent Vanhoucke, et al. “Rethinking the inception architecture for computer vision”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2818–2826.
- [TS14] Alexander Toshev and Christian Szegedy. “Deeppose: Human pose estimation via deep neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 1653–1660.

## References

- [Uij+13] Jasper RR Uijlings et al. “Selective search for object recognition”. In: *International journal of computer vision* 104.2 (2013), pp. 154–171.
- [VML94] Régis Vaillant, Christophe Monrocq, and Yann Le Cun. “Original approach for the localisation of objects in images”. In: *IEE Proceedings-Vision, Image and Signal Processing* 141.4 (1994), pp. 245–250.
- [Wan+12] Tao Wang et al. “End-to-end text recognition with convolutional neural networks”. In: *Pattern Recognition (ICPR), 2012 21st International Conference on*. IEEE. 2012, pp. 3304–3308.
- [WWW18] Zhenhua Wang, Xingxing Wang, and Gang Wang. “Learning fine-grained features via a CNN tree for large-scale classification”. In: *Neurocomputing* 275 (2018), pp. 1231–1240.
- [Xia+14] Tianjun Xiao et al. “Error-driven incremental learning in deep convolutional neural network for large-scale image classification”. In: *Proceedings of the 22nd ACM international conference on Multimedia*. ACM. 2014, pp. 177–186.
- [XT15] Saining Xie and Zhuowen Tu. “Holistically-nested edge detection”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1395–1403.

## References

- [Yan+15] Zhicheng Yan et al. "HD-CNN: hierarchical deep convolutional neural networks for large scale visual recognition". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 2740–2748.
- [Yan+16] Wei Yang et al. "End-to-end learning of deformable mixture of parts and deep convolutional neural networks for human pose estimation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 3073–3082.
- [Yoo+15] Donggeun Yoo et al. "Attentionnet: Aggregating weak directions for accurate object detection". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2659–2667.
- [YP15] Heng Yang and Ioannis Patras. "Mirror, mirror on the wall, tell me, is the error small?" In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 4685–4693.
- [ZF14] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *European conference on computer vision*. Springer. 2014, pp. 818–833.

- [Zha+16] Yu Zhang et al. “Weakly supervised fine-grained categorization with part-based image representation”. In: *IEEE Transactions on Image Processing* 25.4 (2016), pp. 1713–1725.