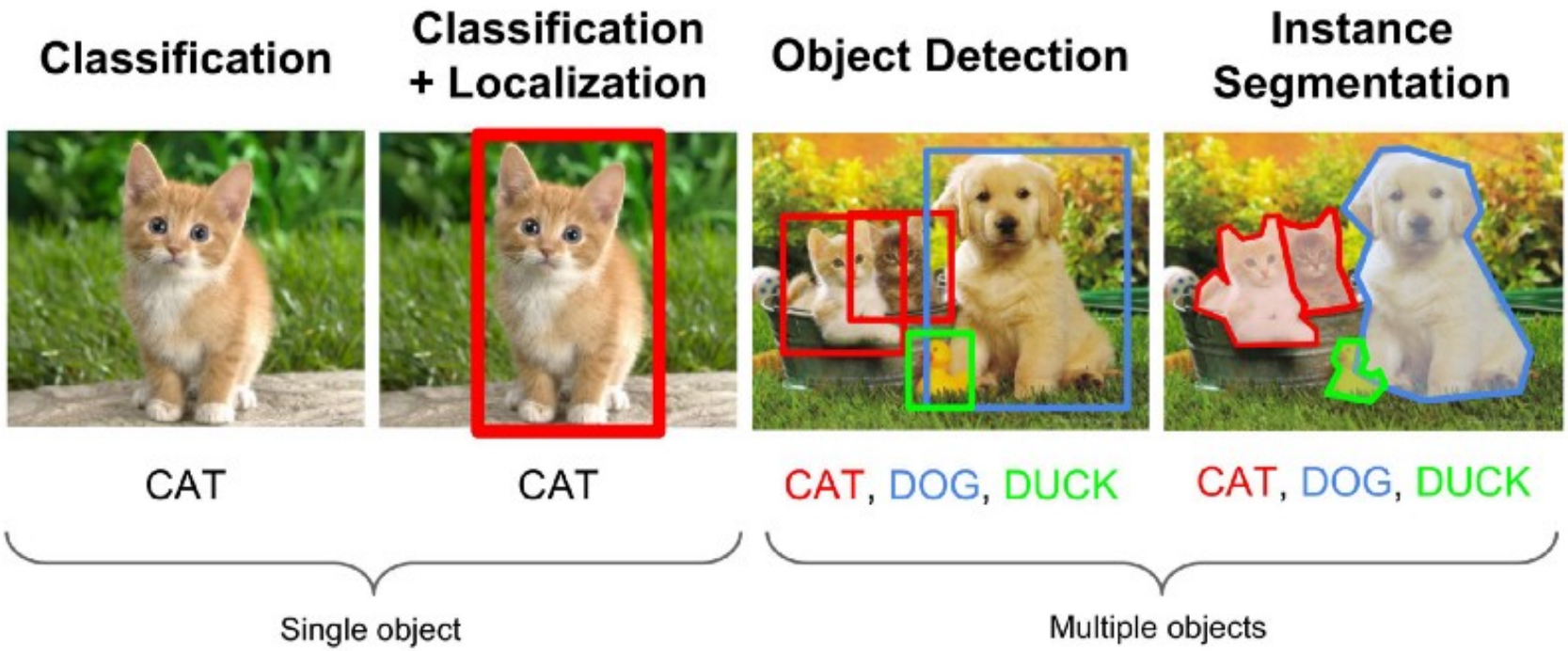
Wykrywanie obiektów

Stanisław Wilczyński

Uniwersytet Wrocławski

11 maja 2018

Analiza obrazków - problemy



Rysunek: Różne problemy dla obrazków

Po co nam wykrywanie obiektów?

- Autonomiczne samochody
- Wykrywanie twarzy (Facebook, aparaty)
- Śledzenie obiektów (automatyczny ruch kamery np. za piłką)
- Liczenie ludzi (badanie ruchu w sklepach, demonstracje, festiwale)
- Podobnie liczenie innych obiektów, np. samochodów
- Visual Search Engine

Na zachętę

Filmik promujący YOLO

Zbiory danych

- PASCAL Visual Object Classification (10 000 obrazków, 20 klas, porządne bounding boxy)
- ImageNet (500 000 obrazków, 200 klas, z bounding boxami)
- Common Objects in COntext (120 000 obrazków, 80 kategorii)

Sposoby oceniania detektorów

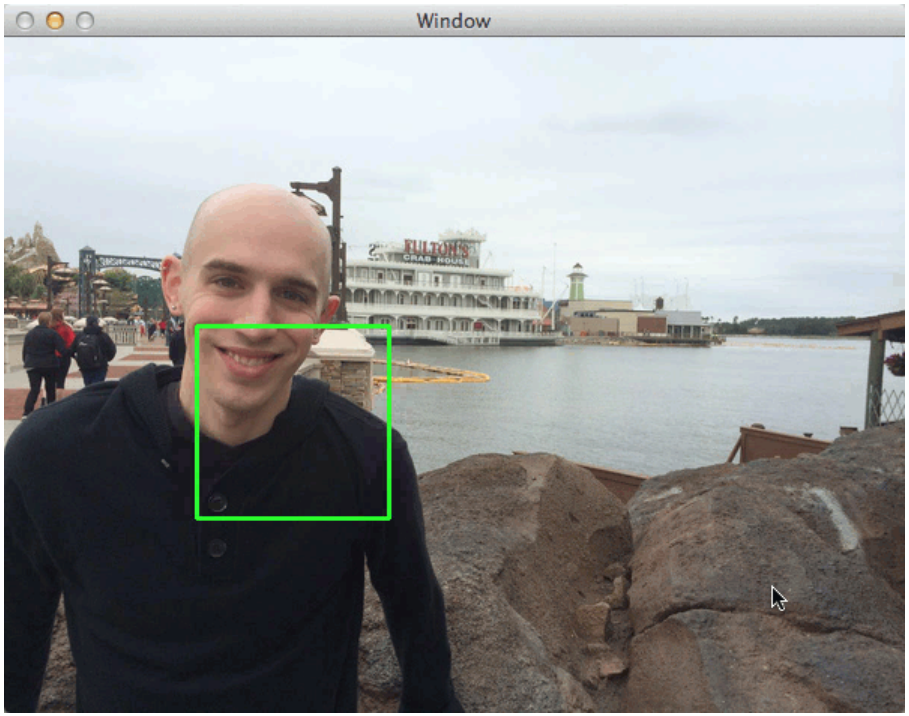
- Intersection over Union (IoU)
- Precision ($\frac{TP}{TP+FP}$) i recall ($\frac{TP}{TP+FN}$)
- TP: dobra klasa, IoU $> t$
- Average Precision

$$AP = \frac{1}{11} \sum_{r \in \{0.0,\ldots,1.0\}} max_{r' \geqslant r} p(r')$$

$p(r)$ – maksymalna precyzja dla zadanego recall

- mean Average Precision (mAP) - średnia z AP po wszystkich klasach

Wczesne metody - okno przesuwne + klasyfikacja

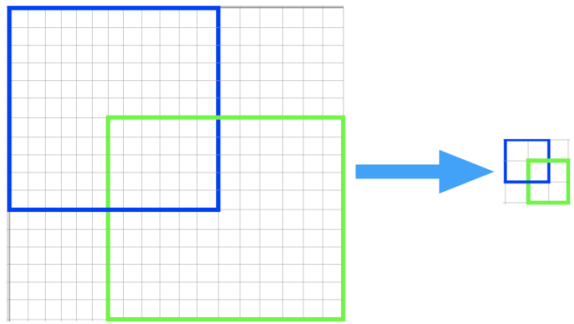


Problemy

- Wielkość okna, obiektów
- Skumulowanie wyników
- Bardzo dużo razy uruchamiany klasyfikator

Pierwsze podejście - OverFeat (2013)

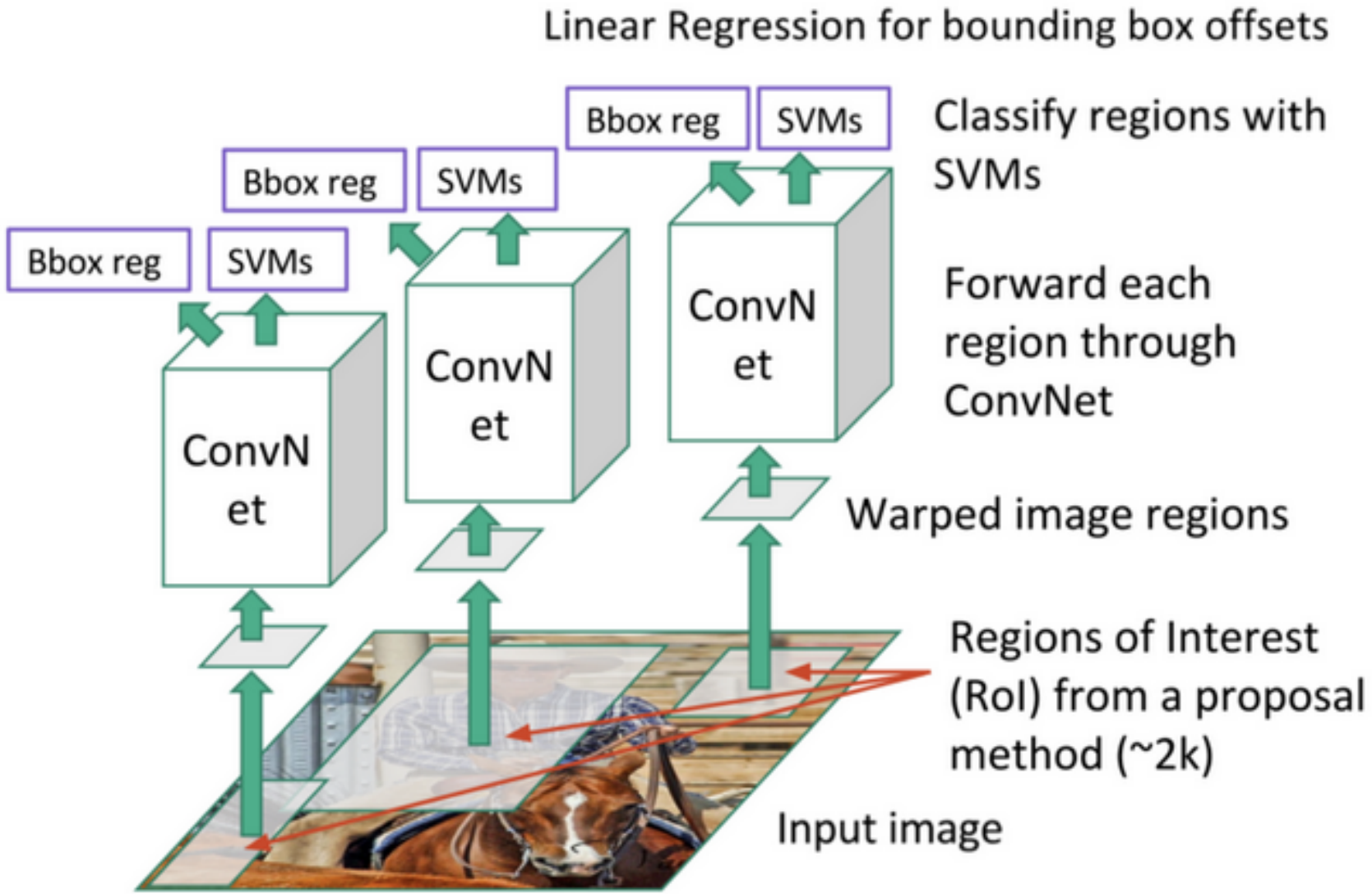
- Okno przesuwne = sploty
- Sieć konwolucyjna do wyciągniecia feature map
- Multiscale classicication (dużo rozmiarów i uśrednienie wyników)



- Do detekcji - dodatkowa klasa tło
- Do lokalizacji - dodatkowe warstwy aplikowane na FM wyznaczające współrzędne i rozmiary (jednego obiektu)
- Łączenie boxów - uśrednianie współrzędnych

Region-based Convolutional Network (R-CNN, 2014)

Motywacja: propozycje regionów



R-CNN

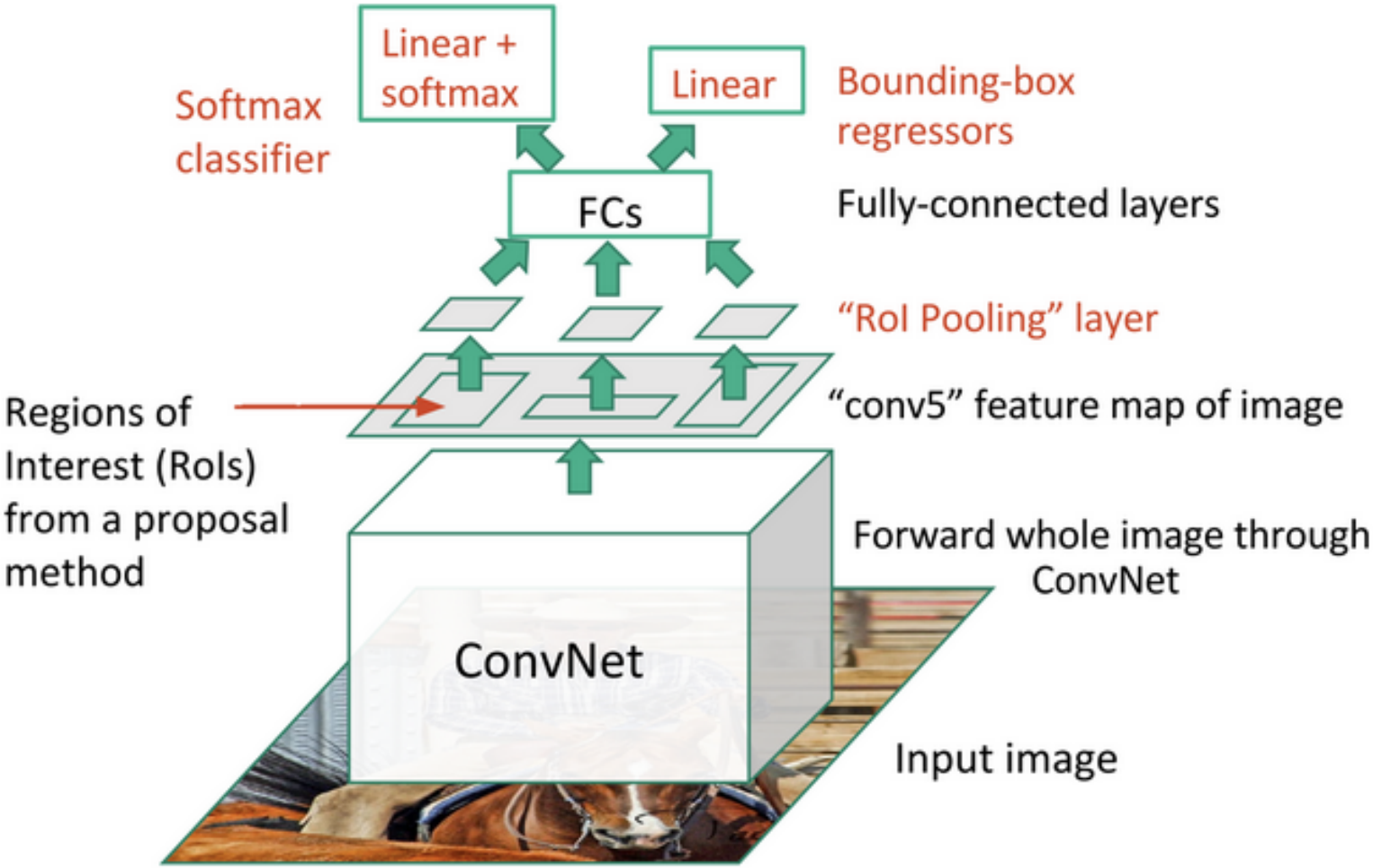
- Selective Search - preprocessing + grupowanie hierarchiczne

- Niezależnie trenowana sieć konwolucyjna do wyciągania feature map
- Małe FM $\Rightarrow$ szybka klasyfikacja za pomocą SVM
- Regresja liniowa dla BB

Fast R-CNN (2015)

Motywacja: przyspieszenie R-CNN

Rozwiązanie: jedno przejście przez sieć konwolucyjną

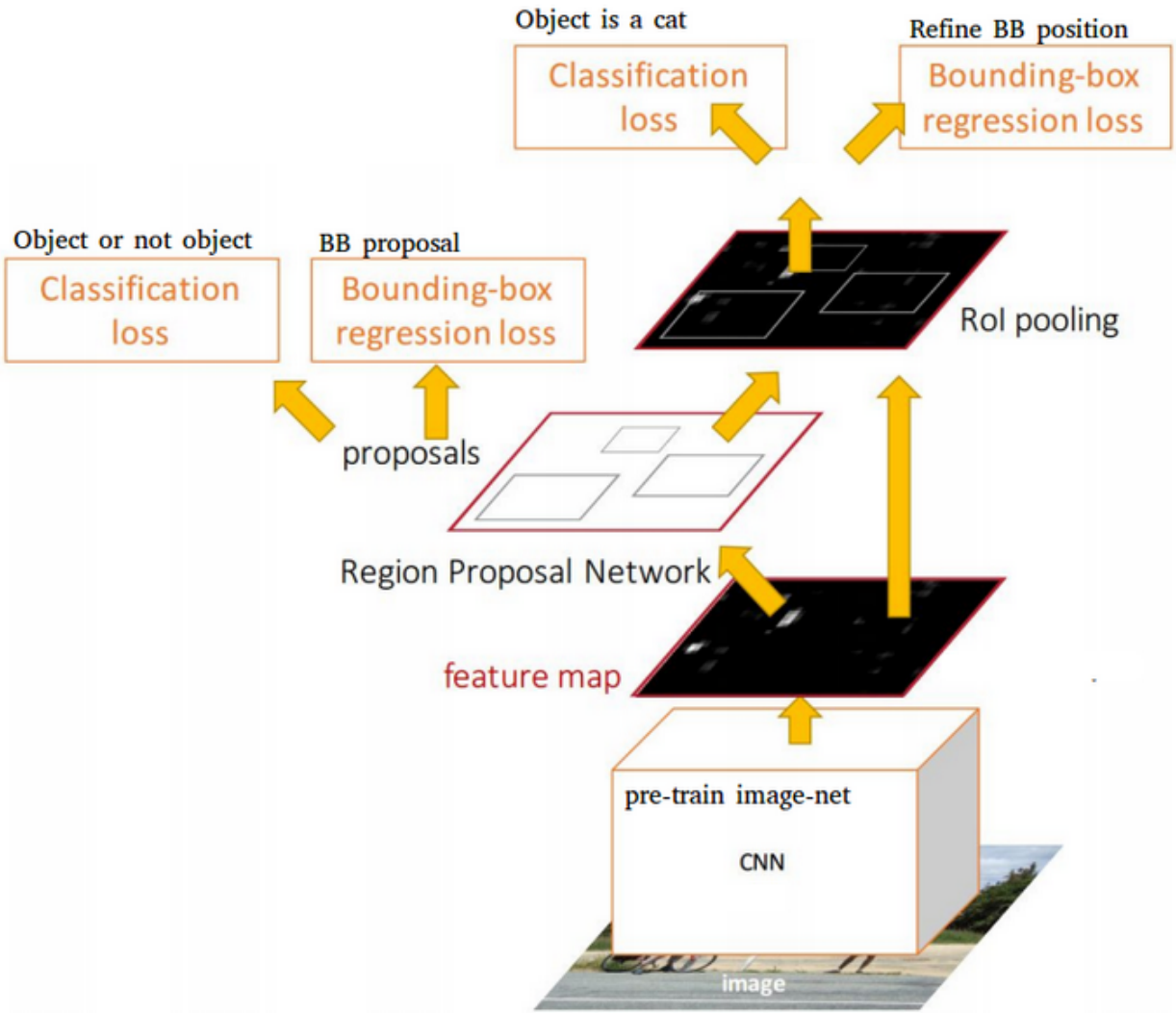


Fast R-CNN

- Propozycje regionów z feature mapy
- Każdy propozycja jest wrzucana do FC oddzielnie
- Roi Pooling - feature mapy regionów do stałego rozmiaru

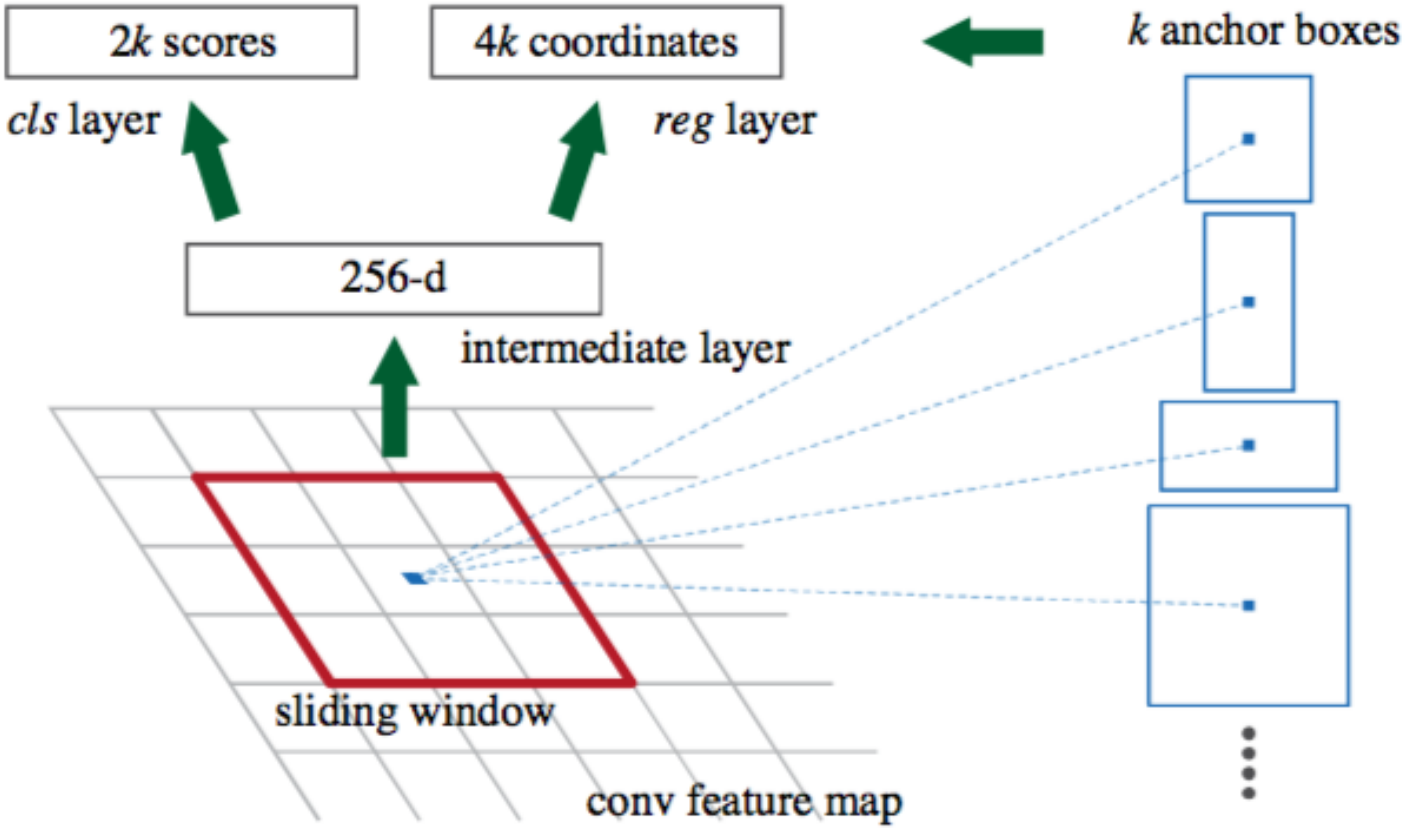
Faster R-CNN (2016)

Motywacja: wyrzućmy wąskie gardło - Selective Search



Faster R-CNN

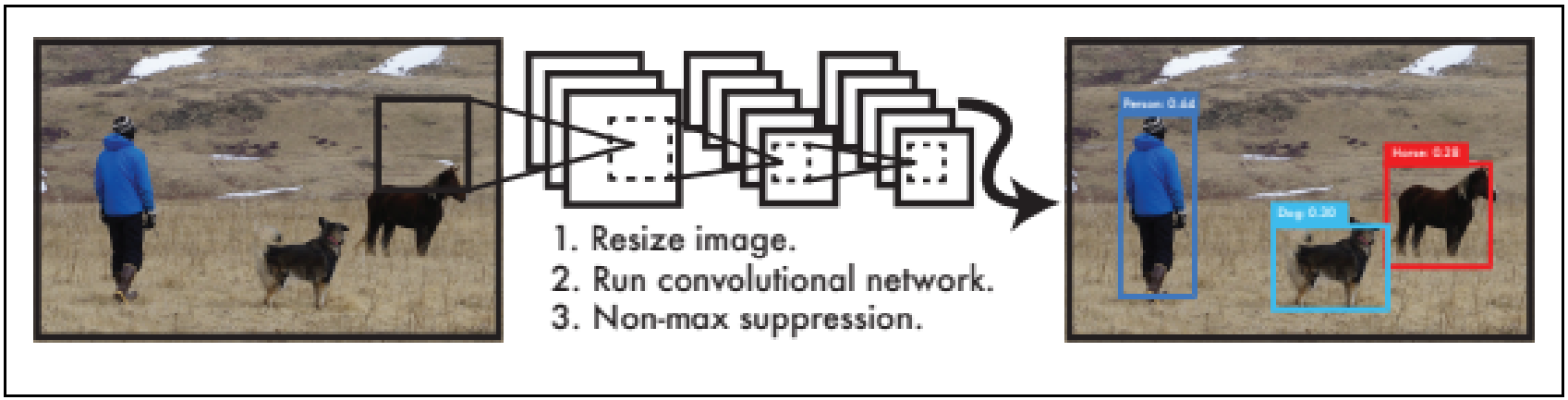
- RPN - max k regionów ze współrzędnymi i scorem
- Przesuwne okno do warstwy FC (ale dzieje się równocześnie)



- RPN + CNN (mechanizm uwagi)

YOLO (2016)

- Bounding boxes i klasyfikacja naraz - jedno przejście przez sieć
- Globalna analiza obrazka

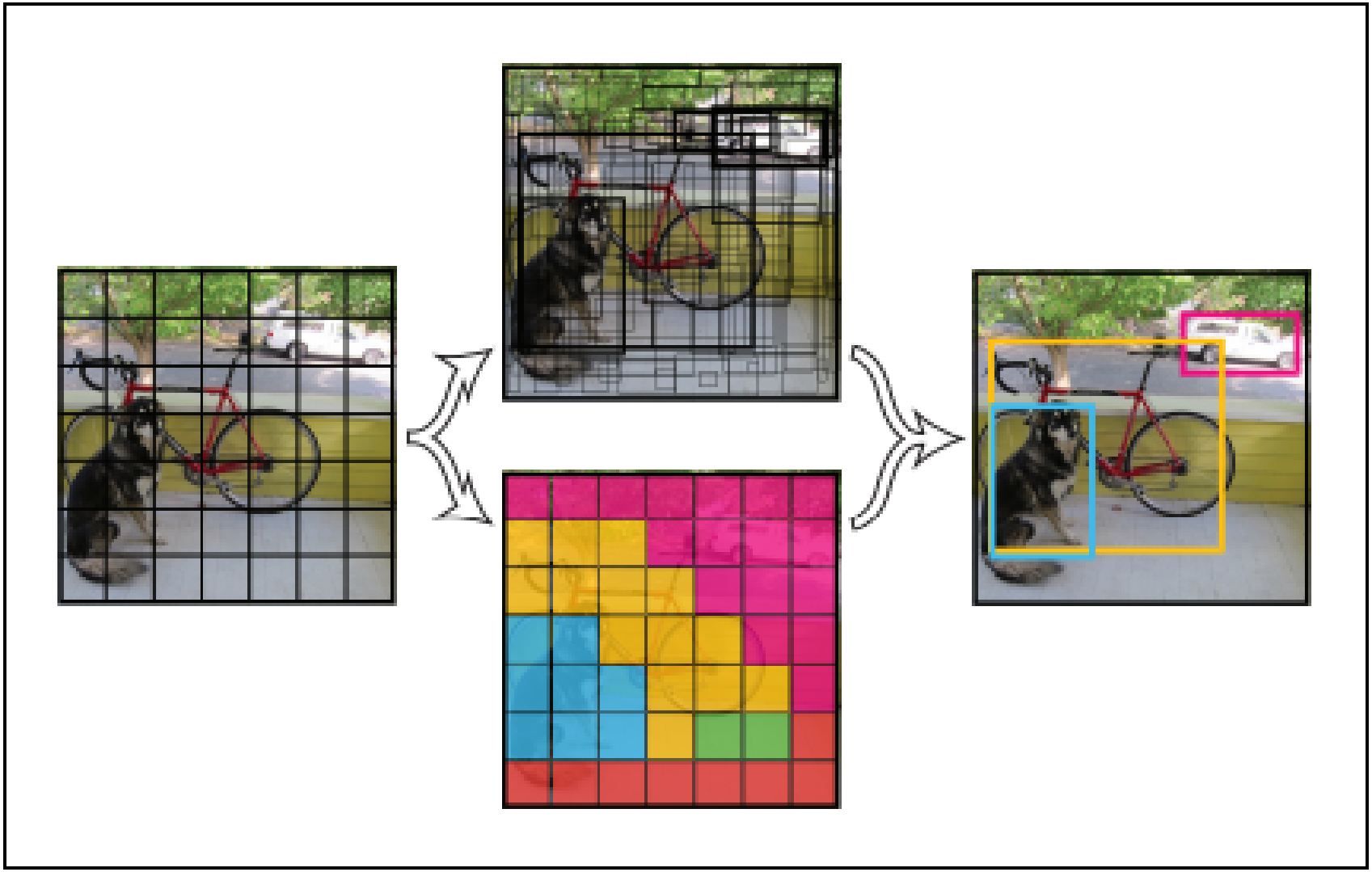


Rysunek: Schemat działania YOLO

Bardziej szczegółowo

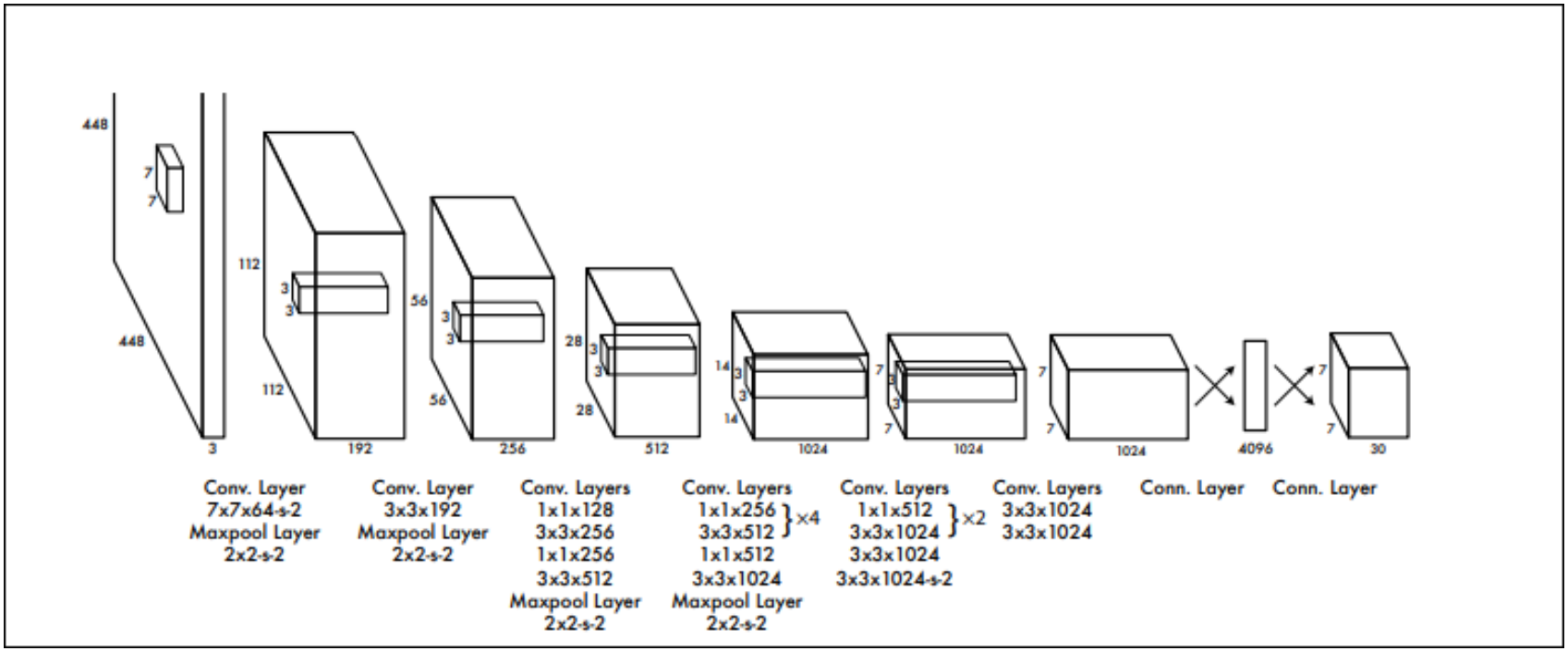
- Siatka $S \times S$
- Jedna komórka $\Rightarrow$ jeden obiekt
- B bounding boxów w każdej komórce
- Box: (x,y,w,h,score)
- Przykład PASCAL VOC: $B = 2, C = 20, S = 7 \Rightarrow$ tensor $7 \times 7 \times 30$

Bounding boxes



Rysunek: Bounding boxes

Architektura sieci



Rysunek: Architektura sieci

Kara

$$\begin{aligned}\text{box confidence score} &\equiv P_r(object) \cdot IoU \\ \text{conditional class probability} &\equiv P_r(class_i|object) \\ \text{class confidence score} &\equiv P_r(class_i) \cdot IoU \\ &= \text{box confidence score} \times \text{conditional class probability}\end{aligned}$$

where

$P_r(object)$ is the probability the box contains an object.
IoU is the IoU (intersection over union) between the predicted box and the ground truth.
$P_r(class_i|object)$ is the probability the object belongs to $class_i$ given an object is presence.
$P_r(class_i)$ is the probability the object belongs to $class_i$

W karze uwzględniamy:

- kara za klasyfikację
- kara za lokalizację
- kara za score boxów

YOLO - generalizacja

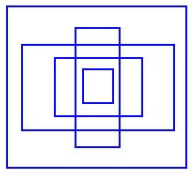
YOLO - generalizacja

Problemy

- Małe obiekty blisko siebie
- Mniejszy mAP niż dla innych metod

YOLOv2 (2016)

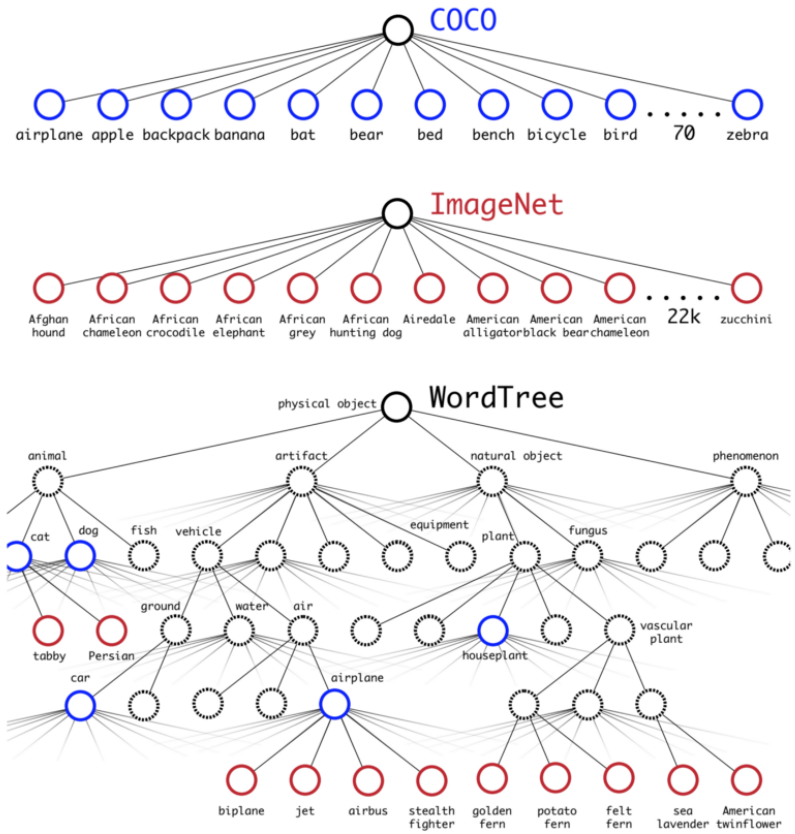
- Batch normalization, zwiększenie rozdzielczości
- Ustalone rozmiary boxów na początku - standardowe kształty
- Anchor boxy wyznaczane k-means z dystansem bazującym na IoU



- Przeniesienie prawdopodobieństw klas do boxów: tensor $(S, S, B \times (5 + C))$, $B = 5, S = 13$
- Fine grained features - wykrywanie mniejszych obiektów za pomocą FM w większej rozdzielczości
- MultiScale Training - co 10 batchów zmieniają rozdzielczości obrazka (można bo same warswy splotowe)
- Zmiana sieci wcześniej trenowanej sieci splotowej

YOLO9000 I

- Połączenie zbiorów danych detekcji i klasyfikacji
- Detekcję i klasyfikację trenujemy oddzielnie
- Problemy: łączenie nazw klas, rozłączne klasy bo softmax



YOLO9000 II

- Błędy klasyfikacji zarówno dla liścia jak i przodków - wyciąga wspólne cechy
- Trenowane na COCO + 9000 klas z ImageNet
- Testowane na ImageNet do detekcji (tylko 44/200 wspólnych klas z COCO)

Inne metody

- Single Shot Detector (2016)
- Neural Architecture Search Net (NASNet, 2017)
- Mask Region-based Convolutional Network (Mask R-CNN, 2017) - również segmentacja obrazka
- RetinaNet (2018)

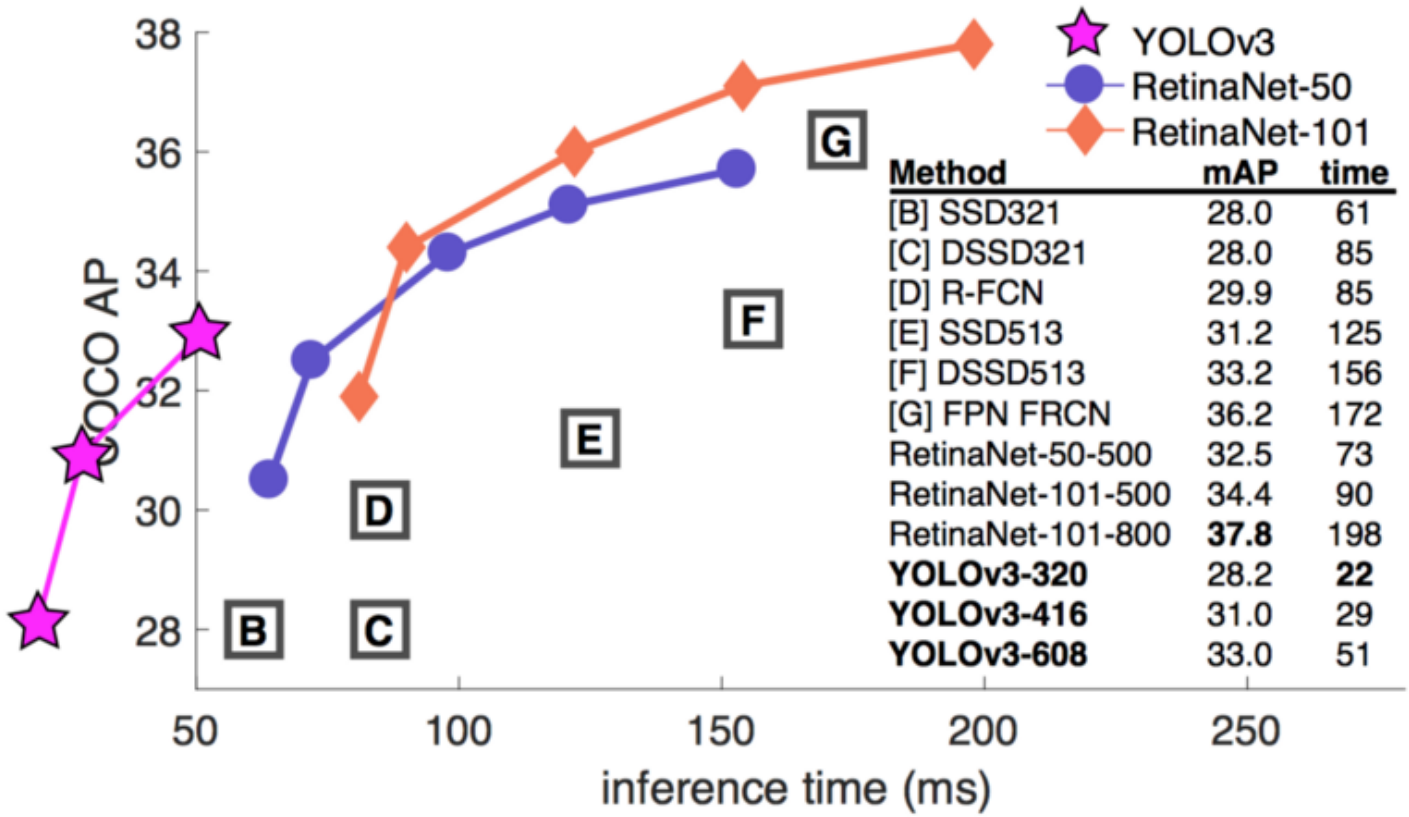
Porównanie działania różnych metod detekcji

Model	PASCAL VOC 2007	PASCAL VOC 2010	PASCAL VOC 2012	COCO 2015 (IoU=0.5)	COCO 2015 (IoU=0.75)	COCO 2015 (Official Metric)	COCO 2016 (IoU=0.5)	COCO 2016 (IoU=0.75)	COCO 2016 (Official Metric)	Real Time Speed
R-CNN	x	62.4%	x	x	x	x	x	x	x	No
Fast R-CNN	70.0%	68.8%	68.4%	x	x	x	x	x	x	No
Faster R-CNN	78.8%	x	75.9%	x	x	x	x	x	x	No
R-FCN	82.0%	x	x	53.2%	x	31.5%	x	x	x	No
YOLO	63.7%	x	57.9%	x	x	x	x	x	x	Yes
SSD	83.2%	x	82.2%	48.5%	30.3%	31.5%	x	x	x	No
YOLOv2	78.6%	x	x	44.0%	19.2%	21.6%	x	x	x	Yes
NASNet	x	x	x	43.1%	x	x	x	x	x	No
Mask R-CNN	x	x	x	x	x	x	62.3%	43.3%	39.8%	No

Rysunek: Porównanie mAP

YOLOv3 (2018)

- Zwiększenie S, B, zmiana sieci splotowej
- Zmiana kary za BB, wiele klas do jednego obiektu
- Feature Pyramid Network (FPN)



Method	mAP	time
[B] SSD321	28.0	61
[C] DSSD321	28.0	85
[D] R-FCN	29.9	85
[E] SSD513	31.2	125
[F] DSSD513	33.2	156
[G] FPN FRCN	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	**37.8**	198
YOLOv3-320	28.2	**22**
YOLOv3-416	31.0	29
YOLOv3-608	33.0	51

Bibliografia I

Ross B. Girshick.

Fast R-CNN.

In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1440–1448, 2015.

Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik.

Region-based convolutional networks for accurate object detection and segmentation.

IEEE Trans. Pattern Anal. Mach. Intell., 38(1):142–158, 2016.

Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi.

You only look once: Unified, real-time object detection.

CoRR, abs/1506.02640, 2015.

Joseph Redmon and Ali Farhadi.

YOLO9000: better, faster, stronger.

CoRR, abs/1612.08242, 2016.

Bibliografia II

Joseph Redmon and Ali Farhadi.

Yolov3: An incremental improvement.

CoRR, abs/1804.02767, 2018.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun.

Faster R-CNN: towards real-time object detection with region proposal networks.

IEEE Trans. Pattern Anal. Mach. Intell., 39(6):1137–1149, 2017.

Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun.

Overfeat: Integrated recognition, localization and detection using convolutional networks.

CoRR, abs/1312.6229, 2013.