Focal Loss for Dense Object Detection

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollár

PAMI 2018, IEEE Transactions on Pattern Analysis and Machine Intelligence

Presenter: Louise Rixon Fuchs (rixon@kth.se)

Outline (3 papers)

$$\mathrm{FL}(p_t) = -(1-p_t)^{\gamma}\log(p_t)$$

- 1. Focal Loss for Dense Object Detection

Work related to the first paper

- 2. Feature Pyramid Networks for Object Detection

- 3. Mask R-CNN

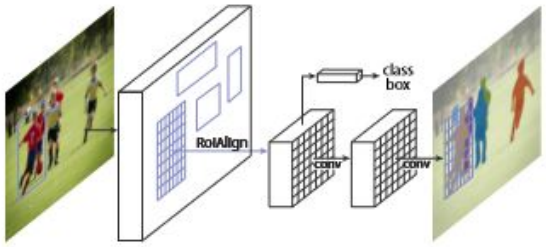


Figure 1. The **Mask R-CNN** framework for instance segmentation.

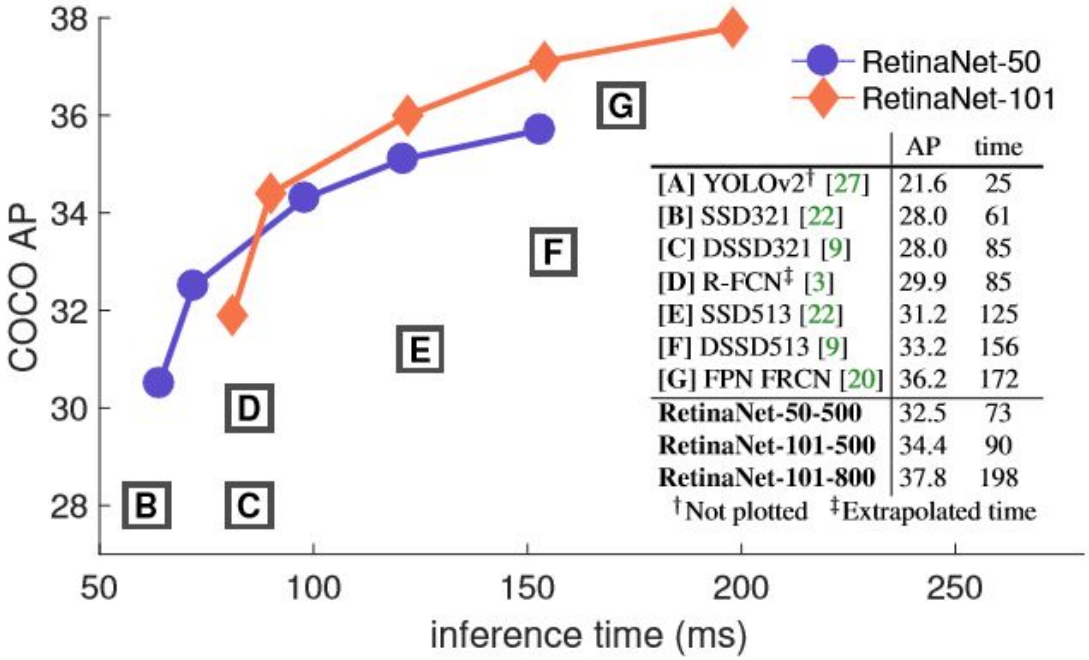
(d) Feature Pyramid Network

Summary Focal Loss

-One stage object detector RetinaNet

-Focal Loss enables to train high-accuracy one-stage detector

- The paper presents a one-stage detector that outperforms state-of-the-art one and two stage detectors



	AP	time
[A] YOLOv2† [27]	21.6	25
[B] SSD321 [22]	28.0	61
[C] DSSD321 [9]	28.0	85
[D] R-FCN‡ [3]	29.9	85
[E] SSD513 [22]	31.2	125
[F] DSSD513 [9]	33.2	156
[G] FPN FRCN [20]	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	37.8	198

†Not plotted ‡Extrapolated time

Figure 2. Speed (ms) versus accuracy (AP) on COCO `test-dev`.

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

Related work

Two stage detectors: R-CNN,

1st stage: generate sparse set of candidate proposals

2nd stage: classifies proposals into classes foreground/background

One-stage detectors: OverFeat, SSD, YOLO

- faster than two stage but “trails in accuracy”
- two stage can be made faster by reducing input images resolution

Aim of the paper, to find out: Can one-stage detectors match or surpass two-stage detectors?

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* .

One stage vs two stage detectors

- State-of-the-art object detectors are based on two-stages ie R-CNN, FPN, Mask R-CNN
- One-stage detector is dense (regular sampling over possible object locations).
- Main obstacle for one-stage detectors for achieving high accuracies is **class imbalance** (classes background and foreground).

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

Class imbalance

- Extreme foreground-background class imbalance
- Imbalance causes two problems:

  - training inefficient since most locations are easy negatives

  - the “mass” of easy negative can lead to degenerative models

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

The solution to class imbalance: Focal Loss

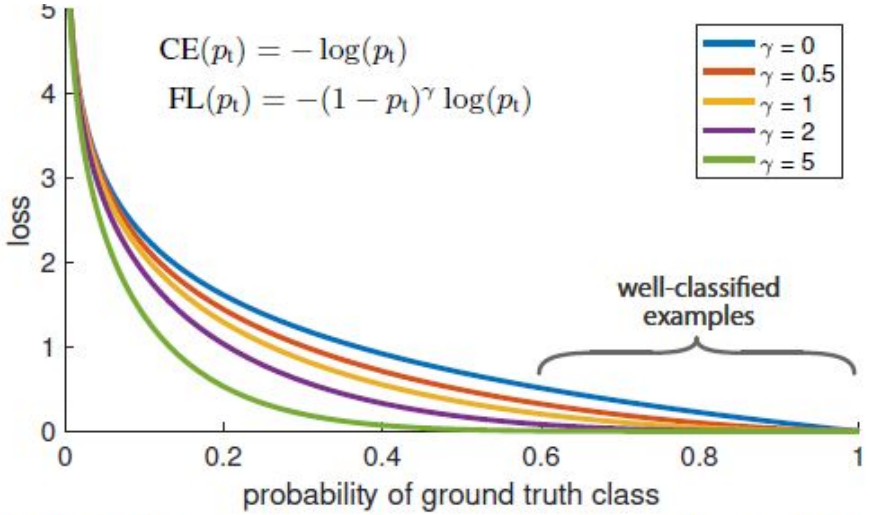
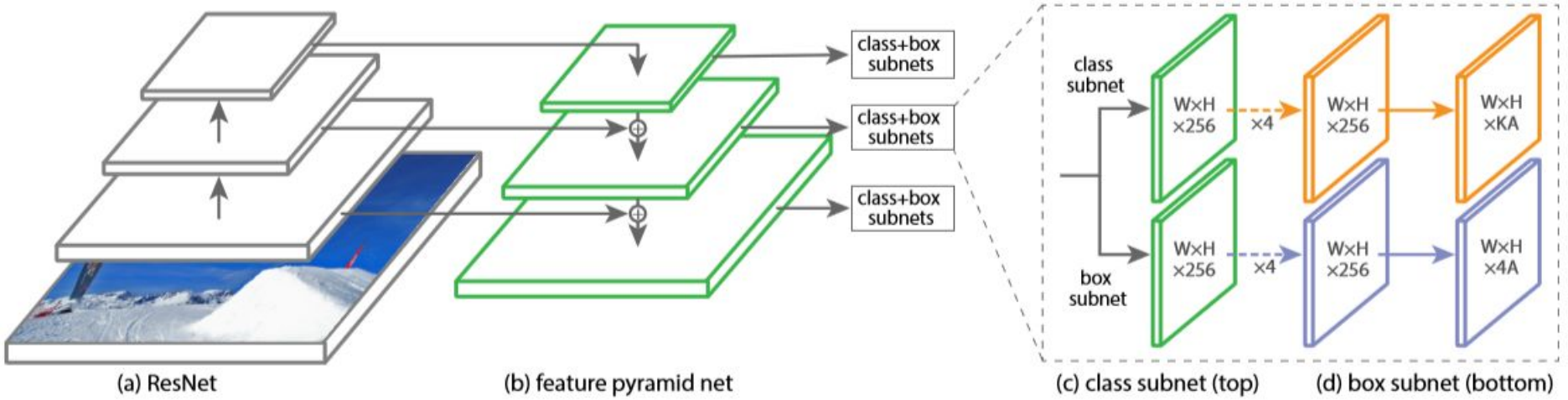


Figure 1: We propose a novel loss we term the *Focal Loss* that adds a factor $(1 - p_t)^\gamma$ to the standard cross entropy criterion. Setting $\gamma > 0$ reduces the relative loss for well-classified examples ($p_t > .5$), putting more focus on hard, misclassified examples. As our experiments will demonstrate, the proposed focal loss enables training highly accurate dense object detectors in the presence of vast numbers of easy background examples.

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

The RetinaNet detector

object classification



multi-scale convolutional feature pyramid

bounding box regression

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

Experiments

1. Training Dense Detections
2. Variants of Focal Loss
3. Model architecture Design
4. Comparison to state of the art

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* .

Experiments (i)

$$\mathrm{FL}(p_t) = -\alpha_t(1-p_t)^{\gamma}\log(p_t)$$

- Training Dense Detection
- One stage detectors fixed sampling grid
- Ablation studies
- New hyperparameter γ introduced, larger γ -> more focus on hard misclassified examples. With γ=0, FL = CE

γ	α	AP	AP_{50}	AP_{75}
0	.75	31.1	49.4	33.0
0.1	.75	31.4	49.9	33.1
0.2	.75	31.9	50.7	33.4
0.5	.50	32.9	51.7	35.2
1.0	.25	33.7	52.0	36.2
2.0	.25	**34.0**	**52.5**	**36.5**
5.0	.25	32.2	49.6	34.8

(b) **Varying γ for FL** (w. optimal α)

method	batch size	nms thr	AP	AP_{50}	AP_{75}
OHEM	128	.7	31.1	47.2	33.2
OHEM	256	.7	31.8	48.8	33.9
OHEM	512	.7	30.6	47.0	32.6
OHEM	128	.5	32.8	50.3	35.1
OHEM	256	.5	31.0	47.4	33.0
OHEM	512	.5	27.6	42.0	29.2
OHEM 1:3	128	.5	31.1	47.2	33.2
OHEM 1:3	256	.5	28.3	42.4	30.3
OHEM 1:3	512	.5	24.0	35.5	25.8
FL	n/a	n/a	**36.0**	**54.9**	**38.7**

(d) **FL *vs.* OHEM** baselines (with ResNet-101-FPN)

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

Experiments (ii)

$$\mathrm{FL}(p_t) = -(1-p_t)^{\gamma}\log(p_t)$$

$$p_t^* = \sigma(\gamma x_t + \beta),$$
$$\mathrm{FL}^* = -\log(p_t^*)/\gamma.$$

- Variants of Focal Loss, FL & FL*
- Expectation that any loss function with similar properties to FL,FL* equally effective.

loss	γ	β	AP	AP_{50}	AP_{75}
CE	–	–	31.1	49.4	33.0
FL	2.0	–	34.0	52.5	36.5
FL*	2.0	1.0	33.8	52.7	36.3
FL*	4.0	0.0	33.9	51.8	36.4

Table 3. Results of FL and FL* versus CE for select settings.

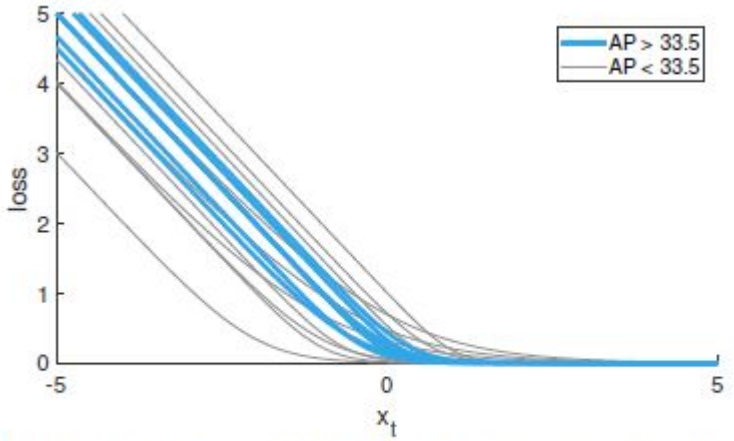


Figure 7. Effectiveness of FL* with various settings γ and β. The plots are color coded such that effective settings are shown in blue.

An example is correctly classified when $x_t > 0,$

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

Experiments (iii)

- Model Architecture Design
  - anchor density, fixed sampling grid, use multiple anchors for high coverage
  - anchor, central point of a sliding window

#sc	#ar	AP	AP_{50}	AP_{75}
1	1	30.3	49.0	31.8
2	1	31.9	50.0	34.0
3	1	31.8	49.4	33.7
1	3	32.4	52.3	33.9
2	3	**34.2**	**53.1**	**36.5**
3	3	34.0	52.5	**36.5**
4	3	33.8	52.1	36.2

(c) **Varying anchor scales and aspects**

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

Experiments (iii)

- Speed versus Accuracy trade-off
- Larger backbone, higher accuracy but slower

depth	scale	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	time
50	400	30.5	47.8	32.7	11.2	33.8	46.1	64
50	500	32.5	50.9	34.8	13.9	35.8	46.7	72
50	600	34.3	53.2	36.9	16.2	37.4	47.4	98
50	700	35.1	54.2	37.7	18.0	39.3	46.4	121
50	800	35.7	55.0	38.5	18.9	38.9	46.3	153
101	400	31.9	49.5	34.1	11.6	35.8	48.5	81
101	500	34.4	53.1	36.8	14.7	38.5	49.1	90
101	600	36.0	55.2	38.7	17.4	39.6	49.7	122
101	700	37.1	56.6	39.8	19.1	40.6	49.4	154
101	800	37.8	57.5	40.8	20.2	41.1	49.2	198

(e) **Accuracy/speed trade-off** RetinaNet (on `test-dev`)

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

Experiments (iv)

- Comparison to State of the Art

	backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Two-stage methods							
Faster R-CNN+++ [16]	ResNet-101-C4	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN w FPN [20]	ResNet-101-FPN	36.2	59.1	39.0	18.2	39.0	48.2
Faster R-CNN by G-RMI [17]	Inception-ResNet-v2 [34]	34.7	55.5	36.7	13.5	38.1	52.0
Faster R-CNN w TDM [32]	Inception-ResNet-v2-TDM	36.8	57.7	39.2	16.2	39.8	**52.1**
One-stage methods							
YOLOv2 [27]	DarkNet-19 [27]	21.6	44.0	19.2	5.0	22.4	35.5
SSD513 [22, 9]	ResNet-101-SSD	31.2	50.4	33.3	10.2	34.5	49.8
DSSD513 [9]	ResNet-101-DSSD	33.2	53.3	35.2	13.0	35.4	51.1
RetinaNet (ours)	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
RetinaNet (ours)	ResNeXt-101-FPN	**40.8**	**61.1**	**44.1**	**24.1**	**44.2**	51.2

Table 2. **Object detection** *single-model* results (bounding box AP), *vs.* state-of-the-art on COCO `test-dev`. We show results for our RetinaNet-101-800 model, trained with scale jitter and for 1.5× longer than the same model from Table 1e. Our model achieves top results outperforming both one-stage and two-stage models. For a detailed breakdown of speed versus accuracy see Table 1e and Figure 2.

Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, .*

Conclusion/Summary

- Class imbalance the primary obstacle preventing one-stage object detectors from surpassing the performance of two-stage methods.
- Class imbalanced is addressed by focus learning on hard negative examples (focal loss is introduced)
- Source code https://github.com/facebookresearch/Detectron

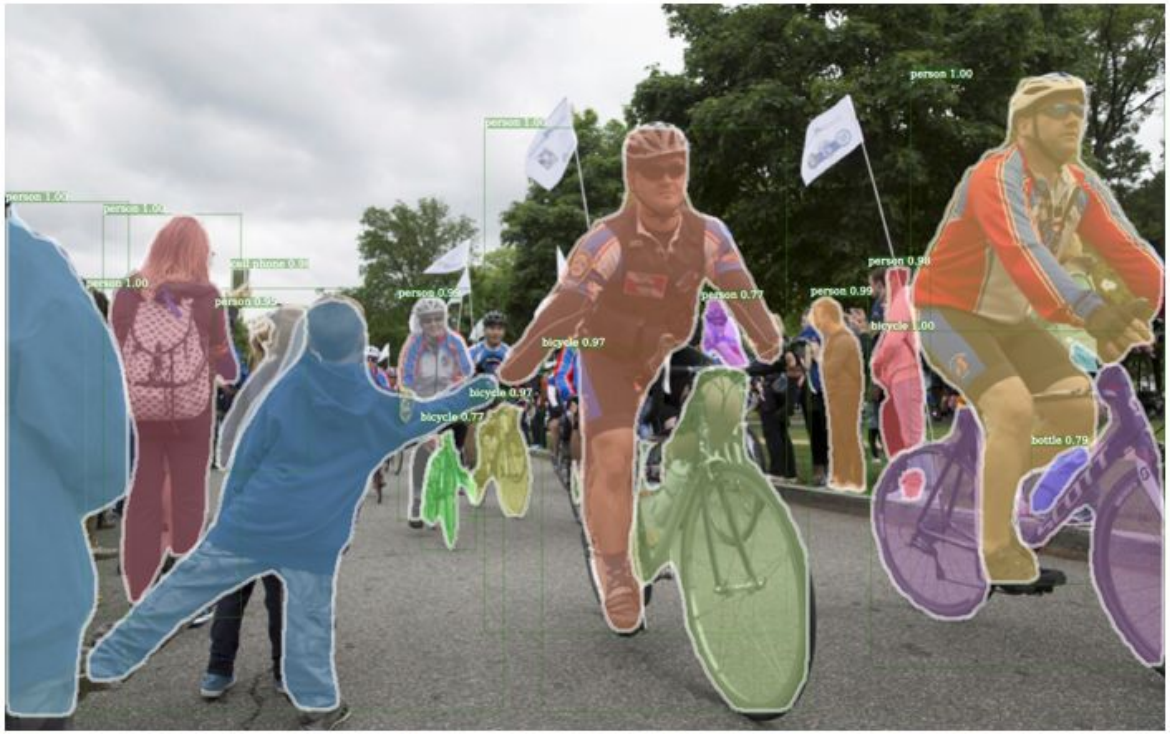
Lin, T., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* .

Related work

- Mask R-CNN
- FPN

Related work: Mask R-CNN

- One framework for: bbox, mask, keypoint



Example Mask R-CNN output.

K. He, G. Gkioxari, P. Doll´ar, and R. Girshick. Mask r-cnn. arXiv:1703.06870, 2017.

Related work: Mask R-CNN

- Extension of Faster R-CNN (+prediction of segmentation masks)
- ROIAlign fixes the pixel-to-pixel alignment between network inputs and outputs.

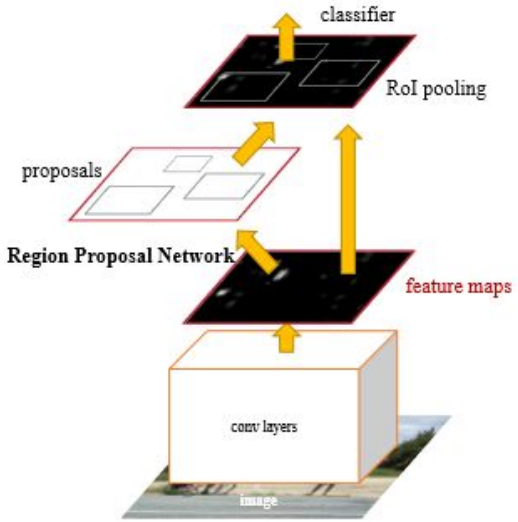


Figure 2: Faster R-CNN is a single, unified network for object detection. The RPN module serves as the 'attention' of this unified network.

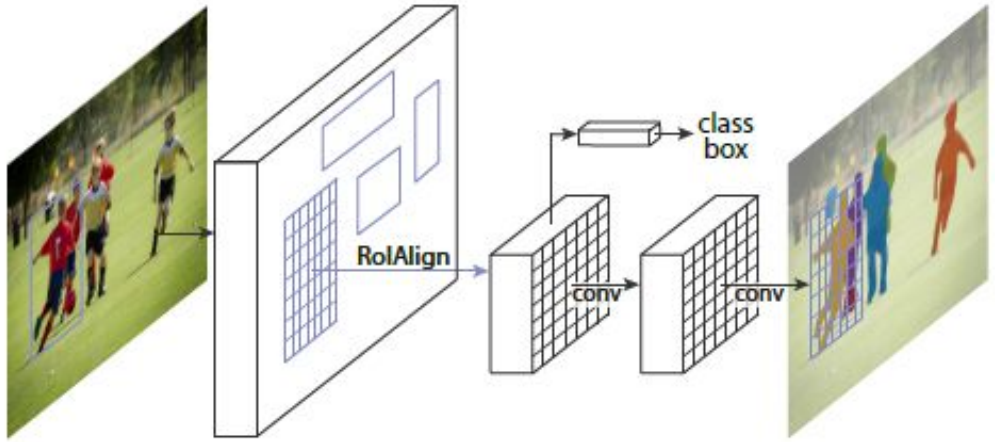


Figure 1. The Mask R-CNN framework for instance segmentation.

K. He, G. Gkioxari, P. Doll´ar, and R. Girshick. Mask r-cnn. arXiv:1703.06870, 2017.

ROIAlign

- ROIPool standard operation for extracting small feature maps, but gives quantizations that cause misalignments
- Can be solved by ROIAlign layer, removes harsh quantization

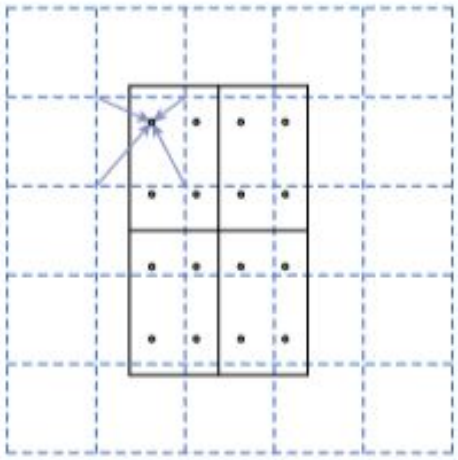


Figure 3. **RoIAlign:** The dashed grid represents a feature map, the solid lines an RoI (with 2×2 bins in this example), and the dots the 4 sampling points in each bin. RoIAlign computes the value of each sampling point by bilinear interpolation from the nearby grid points on the feature map. No quantization is performed on any coordinates involved in the RoI, its bins, or the sampling points.

K. He, G. Gkioxari, P. Doll´ar, and R. Girshick. Mask r-cnn. arXiv:1703.06870, 2017.

Mask R-CNN vs state-of-the-art

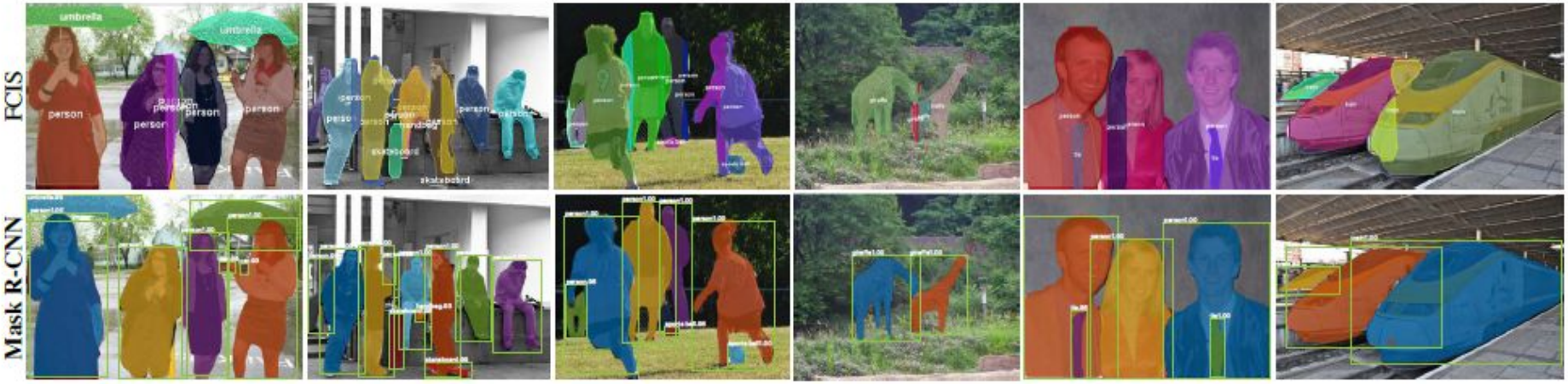
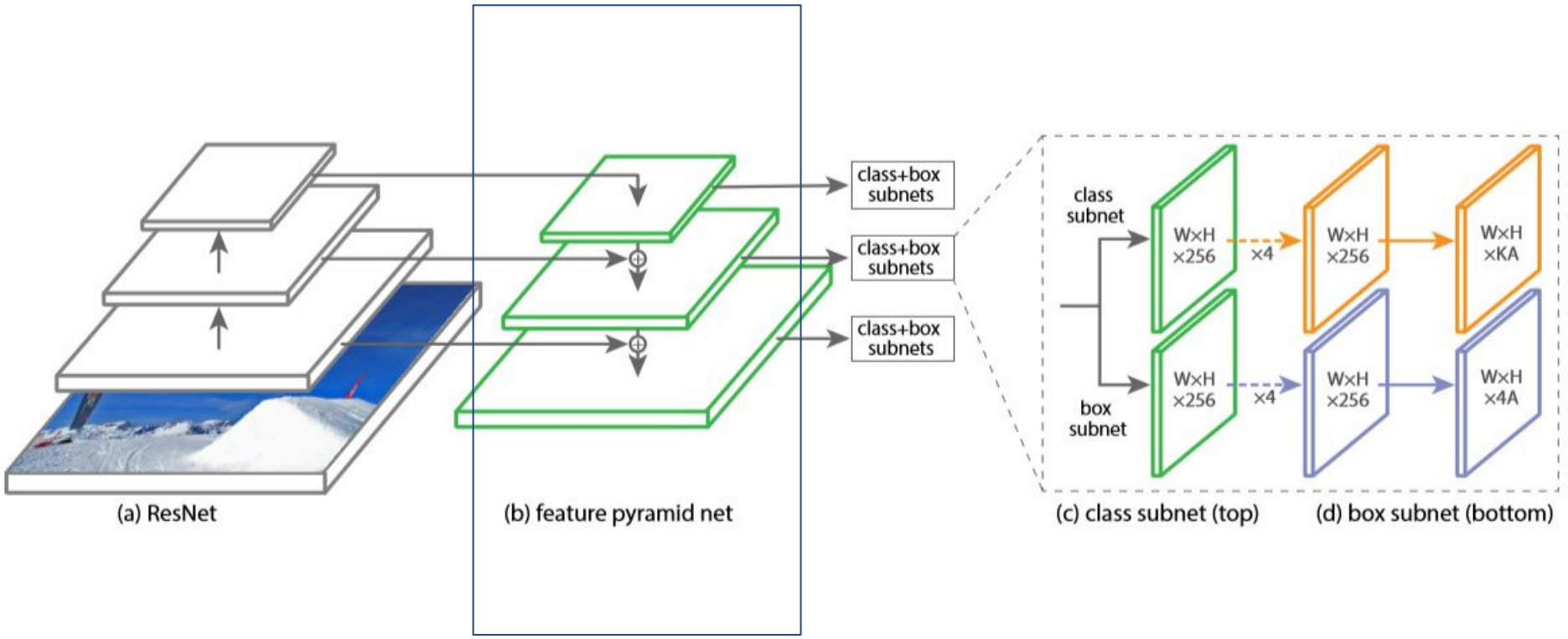


Figure 5. FCIS+++ [21] (top) *vs.* Mask R-CNN (bottom, ResNet-101-FPN). FCIS exhibits systematic artifacts on overlapping objects.

K. He, G. Gkioxari, P. Doll´ar, and R. Girshick. Mask r-cnn. arXiv:1703.06870, 2017.

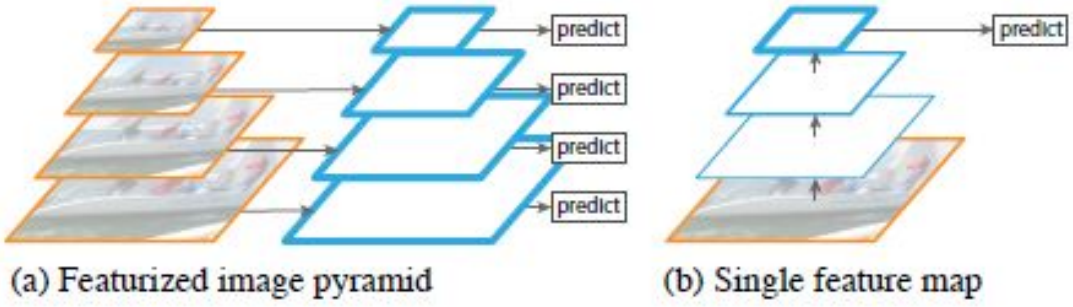
Feature Pyramid Networks for Object Detection



T.-Y. Lin, P. Doll´ar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In CVPR

Feature Pyramid Networks for Object Detection

- Framework for building feature pyramids inside ConvNets.
- Feature pyramid: A basic component in recognition systems for detecting objects at different scales



(a) Featurized image pyramid

(b) Single feature map

ie YOLO

slow

fast but inaccuracte

T.-Y. Lin, P. Doll´ar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In CVPR

Feature Pyramid Networks for Object Detection

- Bottom-up pathway (feature hierarchy, feature maps)
- Top-down pathway & lateral connections (merging feature maps)
- Strong features everywhere!

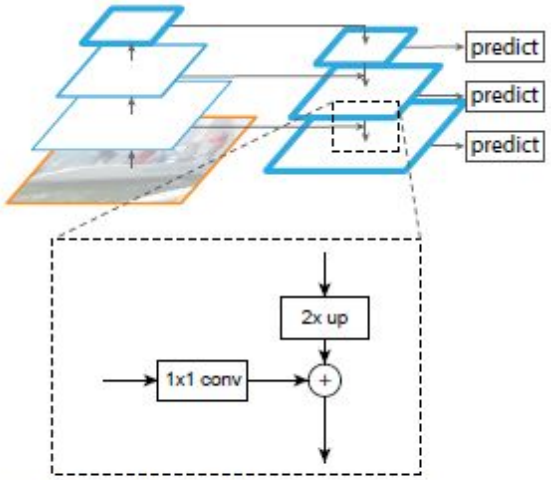


Figure 3. A building block illustrating the lateral connection and the top-down pathway, merged by addition.

T.-Y. Lin, P. Doll´ar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In CVPR

Using FPN for R-CNN

- Results indicate that the feature pyramid is superior to single-scaled features for a region-based object detector.

Fast R-CNN	proposals	feature	head	lateral?	top-down?	AP@0.5	AP	AP_s	AP_m	AP_l
(a) baseline on conv4	RPN, $\{P_k\}$	C_4	conv5			54.7	31.9	15.7	36.5	45.5
(b) baseline on conv5	RPN, $\{P_k\}$	C_5	2*fc*			52.9	28.8	11.9	32.4	43.4
(c) **FPN**	RPN, $\{P_k\}$	$\{P_k\}$	2*fc*	✓	✓	**56.9**	**33.9**	**17.8**	**37.7**	**45.8**
Ablation experiments follow:										
(d) bottom-up pyramid	RPN, $\{P_k\}$	$\{P_k\}$	2*fc*	✓		44.9	24.9	10.9	24.4	38.5
(e) top-down pyramid, w/o lateral	RPN, $\{P_k\}$	$\{P_k\}$	2*fc*		✓	54.0	31.3	13.3	35.2	45.3
(f) only finest level	RPN, $\{P_k\}$	P_2	2*fc*	✓	✓	56.3	33.4	17.3	37.3	45.6

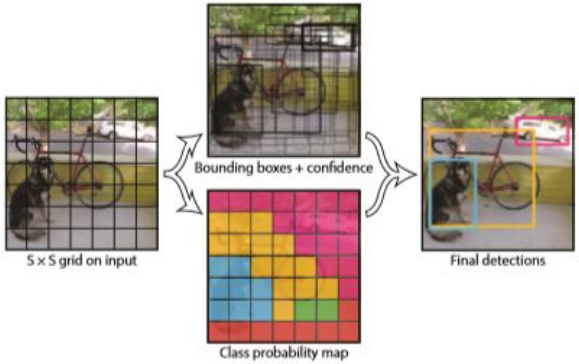
Table 2. Object detection results using **Fast R-CNN** [11] on *a fixed set of proposals* (RPN, $\{P_k\}$, Table 1(c)), evaluated on the COCO `minival` set. Models are trained on the `trainval35k` set. All results are based on ResNet-50 and share the same hyper-parameters.

T.-Y. Lin, P. Doll´ar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In CVPR

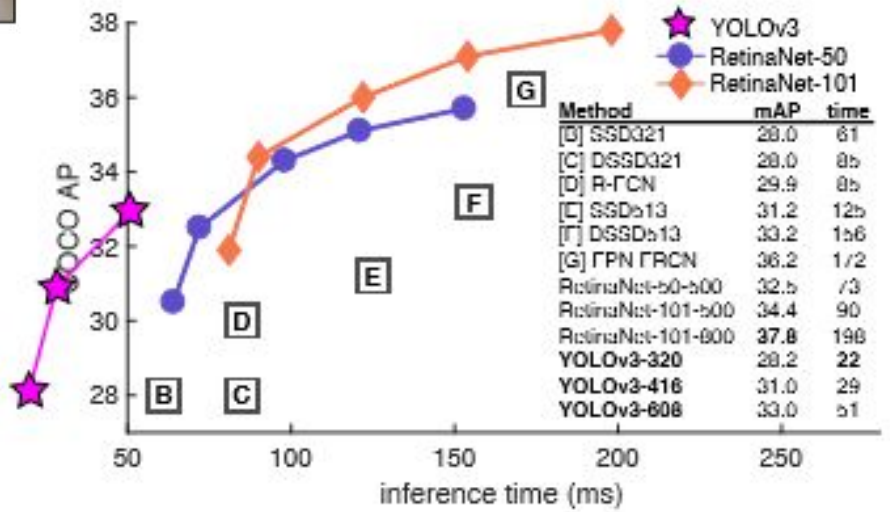
Extra

YOLO: You Only Look Once

- YOLO v1



- YOLO v3 (better than RetinaNet)



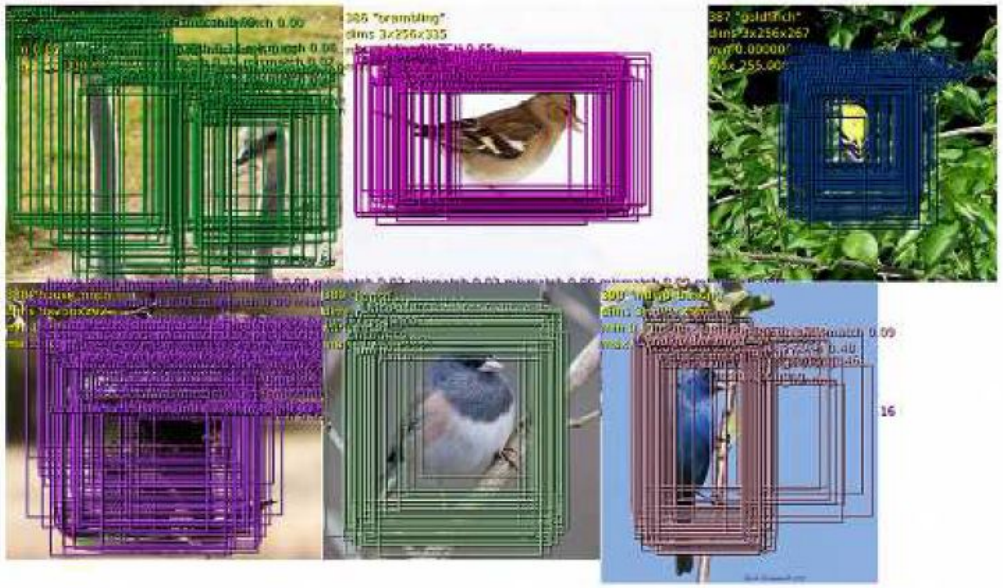
Method	mAP	time
[B] SSD321	28.0	61
[C] DSSD321	28.0	85
[D] R-FCN	29.9	85
[E] SSD513	31.2	125
[F] DSSD513	33.2	156
[G] FPN FRCN	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	**37.8**	198
YOLOv3-320	28.2	**22**
YOLOv3-416	31.0	29
YOLOv3-608	33.0	51

Figure 1. We adapt this figure from the Focal Loss paper [9]. YOLOv3 runs significantly faster than other detection methods with comparable performance. Times from either an M40 or Titan X, they are basically the same GPU.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: CVPR (2016)

OverFeat

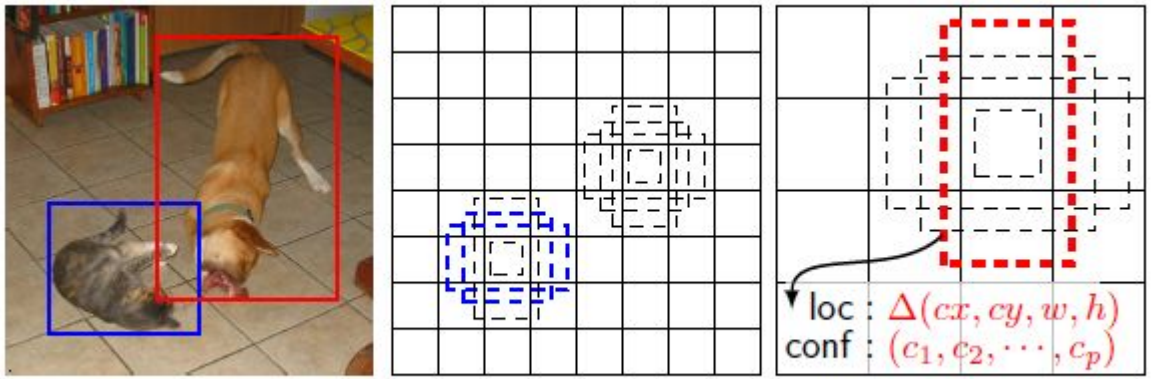
- Integrated approach to object detection, recognition and localization with a single ConvNet
- Accumulating predicted bounding boxes
- Winner of the ILSVRC2013 localization and detection task



P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Le-Cun. Overfeat: Integrated recognition, localization and detectionusing convolutional networks. InICLR, 2014.

SSD: Single Shot Detection

- “Method for detecting objects in an image using a single deep neural network”
- @ prediction time, predicts offset to default boxes. Model loss i weighted sum between L1 and confidence loss (ie Softmax)



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, "SSD: Single shot multibox detector", 2015.

KTH ROYAL INSTITUTE
OF TECHNOLOGY