Feature Pyramid Networks for Object Detection

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie

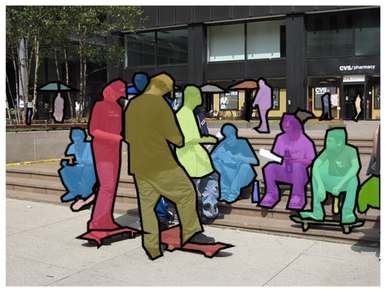
Facebook AI Research

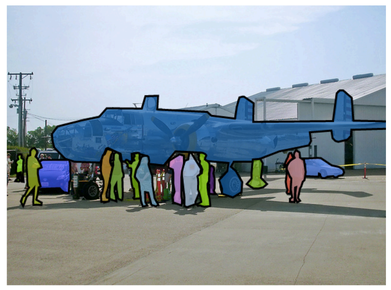
CORNELL TECH

Introduction

Multiscale Object Detection

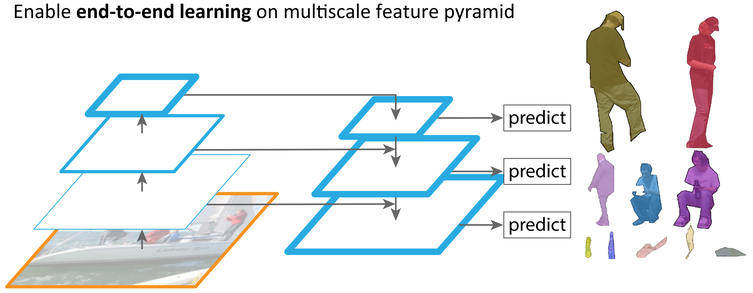
- Goal: learning generic feature representations to detect objects present in multiscale
- Challenges: for objects of all possible scales in the example images,
  - how to learn **semantic strong** multiscale feature representations?
  - how to design **generic features** for various applications in object detection, e.g., object proposals, box localization, instance segmentation?
  - how to compute multiscale feature representations **efficiently**?



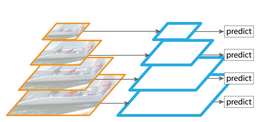


Multiscale Representations

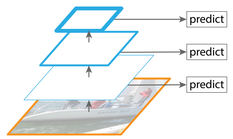
- Leverage **feature hierarchy in ConvNet** to generate semantically strong multiscale feature representations
- Obtain multiscale features from **single forward pass,** adds marginal cost to original bottom-up ConvNet architecture
- Enable **end-to-end learning** on multiscale feature pyramid



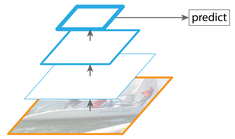
Related Works:



- Pro: semantically strong
- Con: costly to compute



- Pro: fast to compute
- Con: weak features

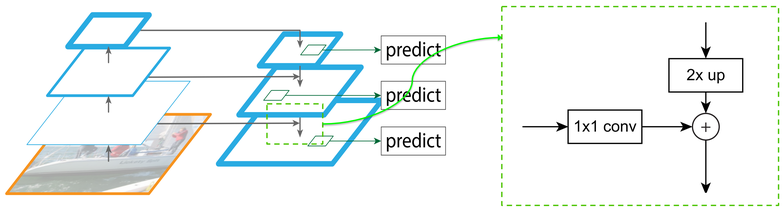


- Pro: Fast to compute
- Con: low res features

Feature Pyramid Networks Architecture

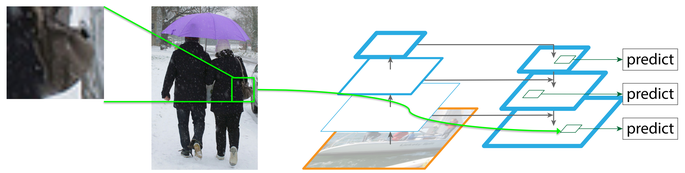
Feature Hallucination

- Simply upsample coarse feature map 2x in **top-down pathway**
- Add missing high resolution information from **lateral connections**



Why does FPN Improves Features for Small Objects?

- FPN leverages **contextual information** passed top-down for small objects
- FPN increases **feature resolution** for small objects



Fast R-CNN + FPN

- Regions to feature Levels: $k = \lfloor k_0 + \log_2(\sqrt{wh}/224) \rfloor$
- **+ 2.3 AP** (over Fast R-CNN baseline)

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	2fc			52.9	28.8	11.9	32.4	43.4
(c) **FPN**	RPN, $\{P_k\}$	$\{P_k\}$	2fc	✓	✓	**56.9**	**33.9**	**17.8**	**37.7**	**45.8**

Strong Features Enable Efficient Learning

	Fast R-CNN + FPN	Fast R-CNN
Feature dimension	256	1024
Head Classifier	2-mlp	conv5
Training Time	10.6 hr	44.6 hr
Inference Time	0.15 s	0.32 s
Accuracy	33.9 AP	31.9 AP

State-of-the-art on COCO Leaderboard

- Best single model accuracy
- Runs at 0.172s per image (5-6 fps)

method	backbone	competition	image pyramid	test-dev $AP_{@.5}$	test-dev AP	test-dev AP_s	test-dev AP_m	test-dev AP_l	test-std $AP_{@.5}$	test-std AP	test-std AP_s	test-std AP_m	test-std AP_l
ours, Faster R-CNN on **FPN**	ResNet-101	-		**59.1**	**36.2**	**18.2**	**39.0**	48.2	**58.5**	**35.8**	**17.5**	**38.7**	47.8
*Competition-winning **single-model** results follow:*													
G-RMI†	Inception-ResNet	2016		-	34.7	-	-	-	-	-	-	-	-
AttractioNet‡ [3]	VGG16 + Wide ResNet§	2016	✓	53.4	35.7	15.6	38.0	**52.7**	52.9	35.3	14.7	37.6	**51.9**
Faster R-CNN +++ [4]	ResNet-101	2015	✓	55.7	34.9	15.6	38.7	50.9	-	-	-	-	-
Multipath [9] (on minival)	VGG-16	2015		49.6	31.5	-	-	-	-	-	-	-	-
ION‡ [1]	VGG-16	2015		53.4	31.2	12.8	32.9	45.2	52.9	30.7	11.8	32.8	44.8

Region Proposal + FPN

- Anchors to feature Levels: {32^2, 64^2, 128^2, 256^2, 512^2} → {P_2, P_3, P_4, P_5, P_6}
- **+ 7.9 AR** (over original RPN)
- **+12.9 AR** (for small objects)

RPN	feature	# anchors	lateral?	top-down?	AR^{100}	AR^{1k}	AR^{1k}_s	AR^{1k}_m	AR^{1k}_l
(a) baseline on conv4	C_4	47k			36.1	48.3	32.0	58.7	62.2
(b) baseline on conv5	C_5	12k			36.3	44.9	25.3	55.5	64.2
(c) **FPN**	$\{P_k\}$	200k	✓	✓	**44.0**	**56.3**	**44.9**	**63.4**	66.2

How important are top-down and lateral connections?

- **+ 6.6 AR** (over only bottom-up pyramid)
- **+ 9.5 AR** (over only top-down pyramid)
- **+ 5.6 AR** (over only finest level)

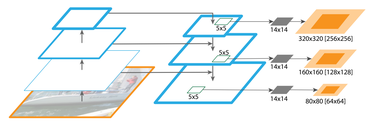
Ablation experiments follow:

(c) **FPN**	$\{P_k\}$	200k	✓	✓	**44.0**	**56.3**	**44.9**	**63.4**	66.2
(d) bottom-up pyramid	$\{P_k\}$	200k	✓		37.4	49.5	30.5	59.9	**68.0**
(e) top-down pyramid, w/o lateral	$\{P_k\}$	200k		✓	34.5	46.1	26.5	57.4	64.7
(f) only finest level	P_2	750k	✓	✓	38.4	51.3	35.1	59.7	67.6

DeepMask + FPN

- **+ 11.0 AR** (over DeepMask baseline)
- End-to-end instance segmentation without image pyramid
- State-of-the-art for both performance and speed

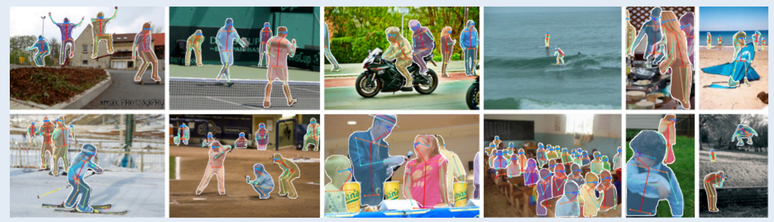
	image pyramid	AR	AR_s	AR_m	AR_l	time (s)
DeepMask [5]	✓	37.1	15.8	50.1	54.9	0.49
SharpMask [6]	✓	39.8	17.4	53.1	59.1	0.77
InstanceFCN [2]	✓	39.2	–	–	–	1.50†
FPN Mask Results:						
DeepMask + FPN		**48.1**	**32.6**	**54.2**	**65.6**	0.25



Recent Use: Mask R-CNN

Mask R-CNN, Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick, arXiv 2017

- Leverage FPN multiscale feature representations
- Light weight features and predictors allow flexible design for multitask learning
- State-of-the-art on instance segmentation and person keypoints localization



Summary

- Enable end-to-end learning with ConvNet feature pyramid representations
- Efficiently compute strong features by leveraging feature hierarchy in ConvNet
- Strong feature representations improve results on all tested tasks
- Generic feature representations applicable for various object detection applications

tl483@cornell.edu