ECCV 2018 European Conference on Computer Vision
8 – 14 September 2018 | Munich, Germany

COCO Common Objects in Context

Hybrid Task Cascade for Instance Segmentation

Kai Chen[1], Jiangmiao Pang[2,3], Jiaqi Wang[1], Yu Xiong[1], Xiaoxiao Li[1], Shuyang Sun[4], Wansen Feng[2]
Ziwei Liu[1], Jianping Shi[2], Wanli Ouyang[4], Chen Change Loy[1,5], Dahua Lin[1]

[1]The Chinese University of Hong Kong [2]SenseTime Research [3]Zhejiang University
[4]The University of Sydney [5]Nanyang Technological University of Singapore

Team: MMDet

香港中文大學
The Chinese University of Hong Kong

商汤 sensetime

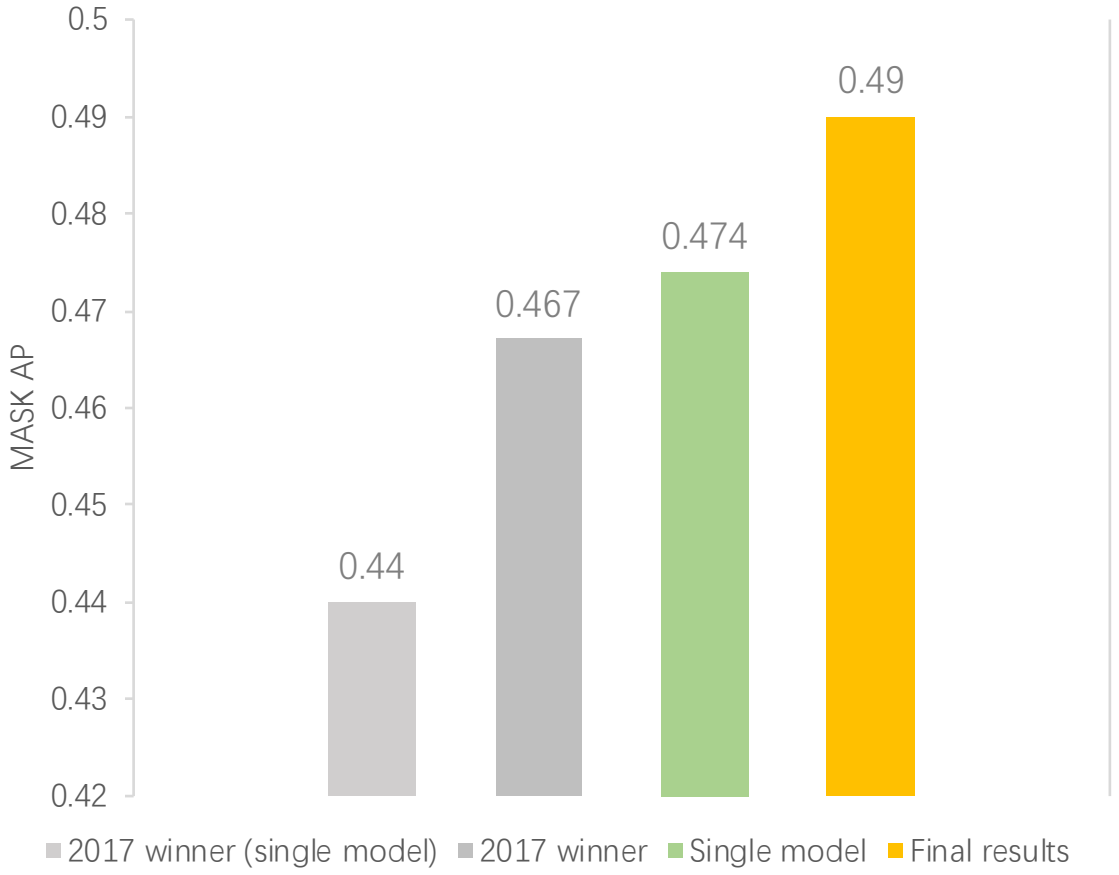
浙江大學 ZHEJIANG UNIVERSITY

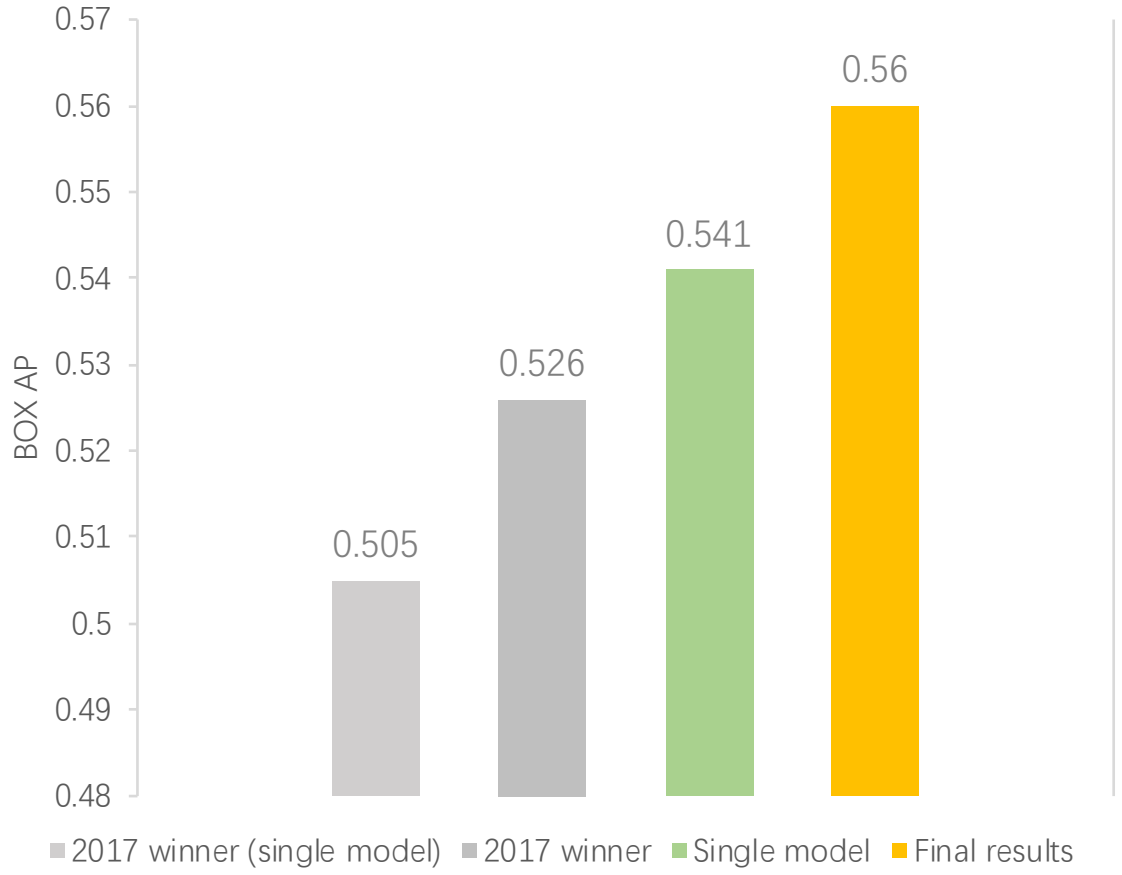
THE UNIVERSITY OF SYDNEY

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Results

Comparison of our approach with 2017 winning entries on COCO test-dev.





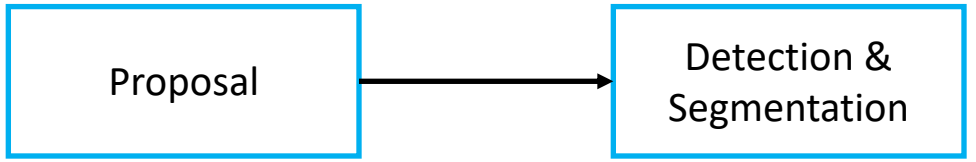
Overview

1. We developed a **hybrid cascading and branching** pipeline for detection and segmentation.

Detection &
Segmentation

Overview

1. We developed a **hybrid cascading and branching** pipeline for detection and segmentation.
2. We proposed a **feature guided anchoring** scheme to improve the average recall (AR) of RPN by 10 points. (submitted to AAAI 2019)



Overview

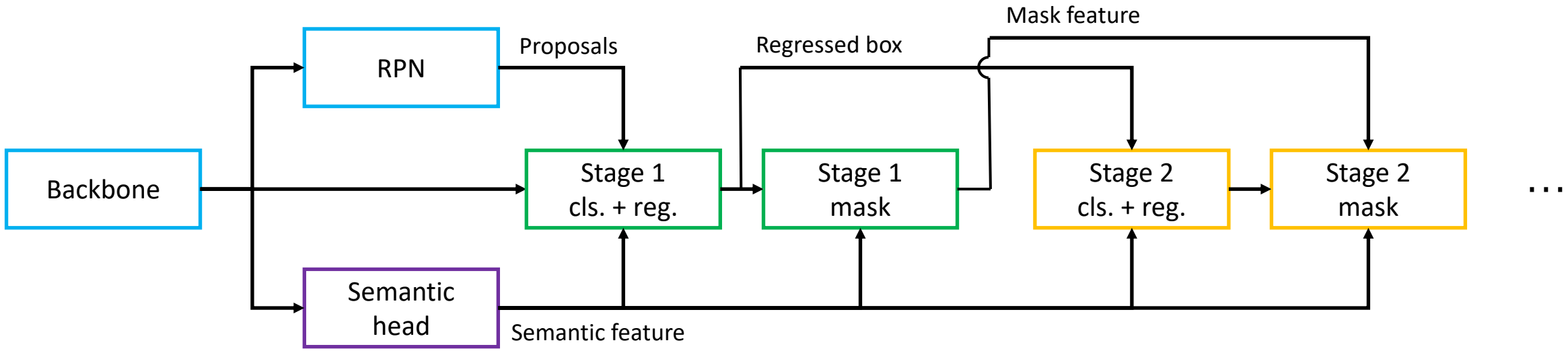
1. We developed a **hybrid cascading and branching** pipeline for detection and segmentation.
2. We proposed a **feature guided anchoring** scheme to improve the average recall (AR) of RPN by 10 points. (submitted to AAAI 2019)
3. We designed a new backbone **FishNet**. (accepted to NIPS 2018)

Overview

1. We developed a **hybrid cascading and branching** pipeline for detection and segmentation.
2. We proposed a **feature guided anchoring** scheme to improve the average recall (AR) of RPN by 10 points. (submitted to AAAI 2019)
3. We designed a new backbone **FishNet**. (accepted to NIPS 2018)

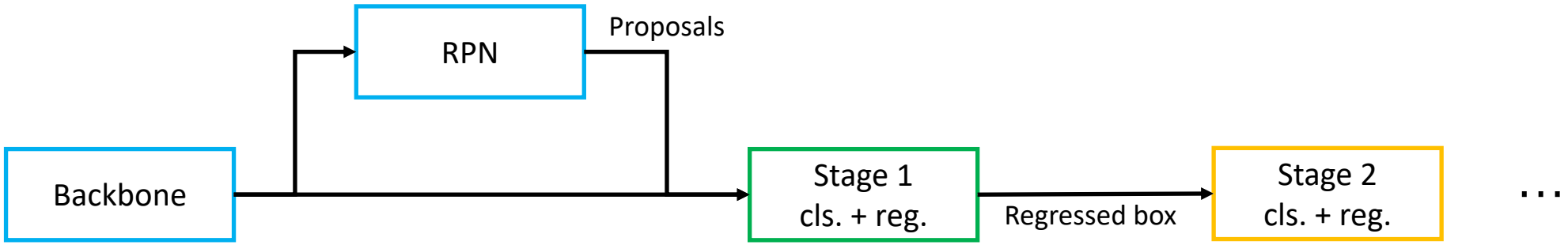
Pipeline

A hybrid architecture with interleaved task branching and cascade.



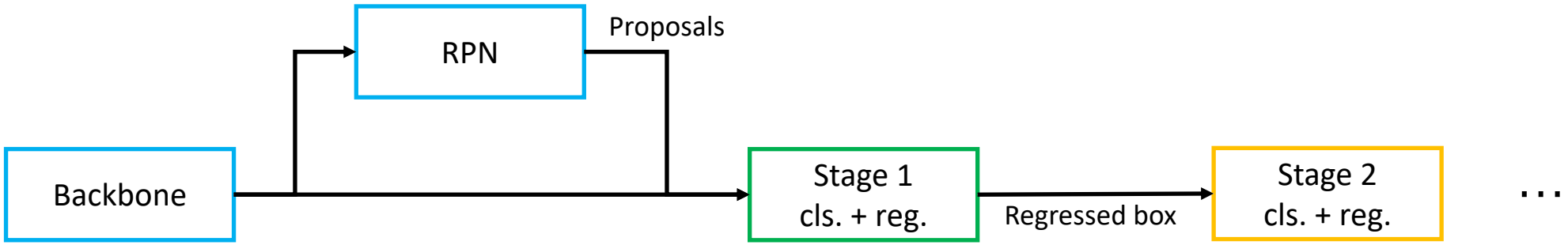
Pipeline

Baseline: Cascade R-CNN



Pipeline

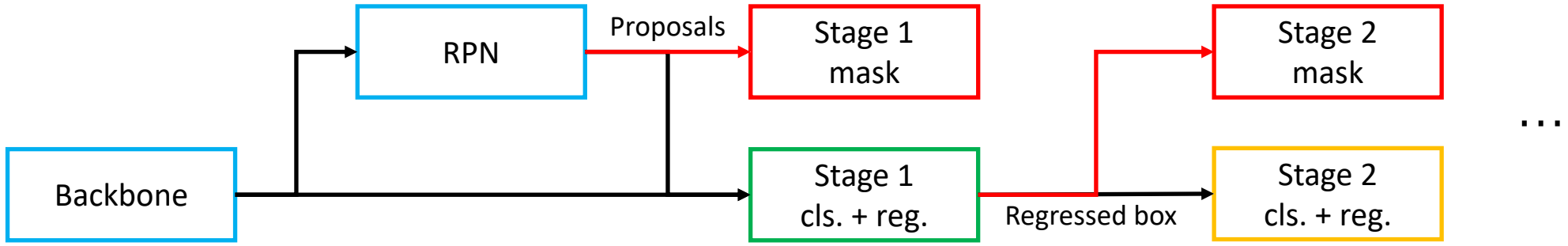
Baseline: Cascade R-CNN



Problem: designed for detection, not segmentation

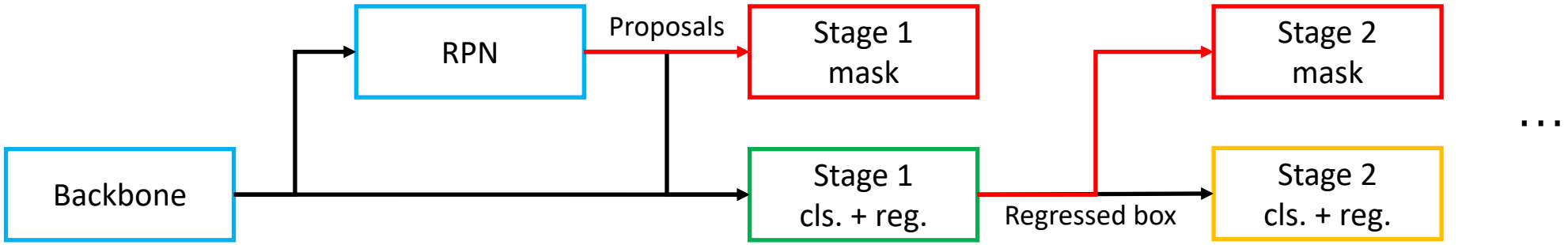
Pipeline

Baseline: Cascade R-CNN + Mask R-CNN



Pipeline

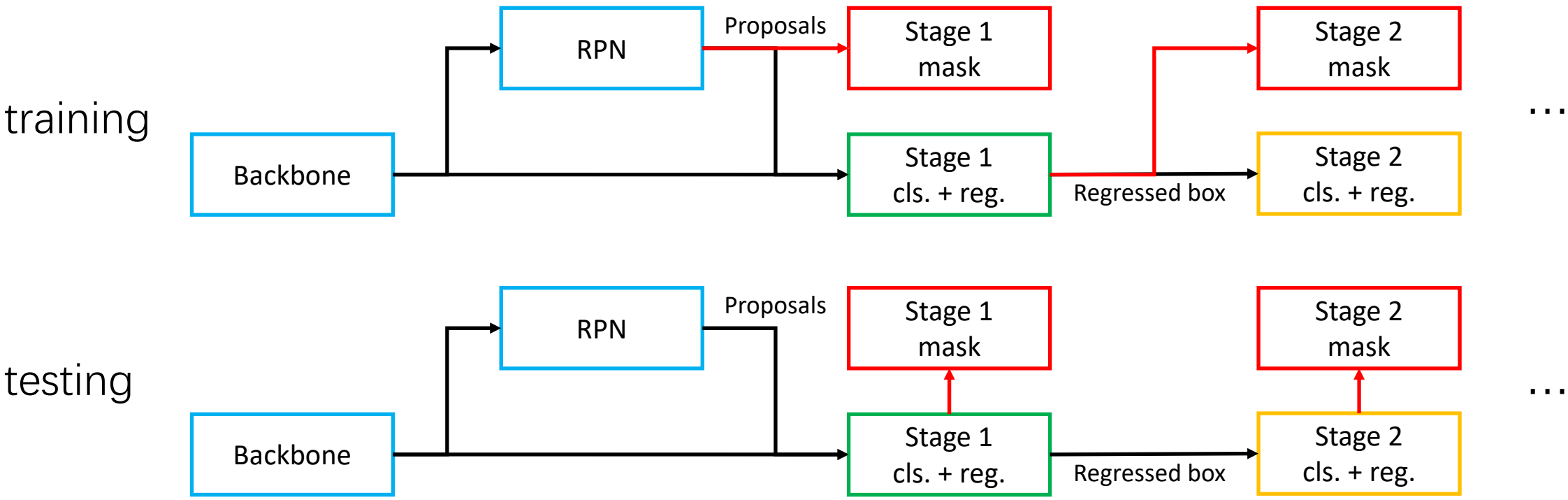
Baseline: Cascade R-CNN + Mask R-CNN



Problem: mismatch of training and testing pipeline

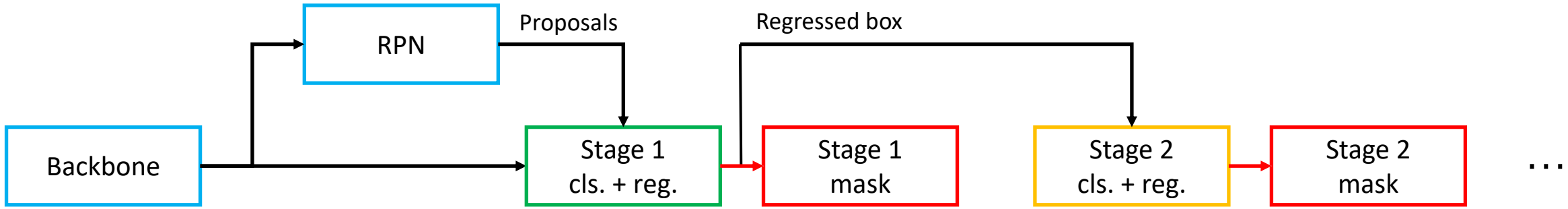
Pipeline

Problem: mismatch of training and testing pipeline



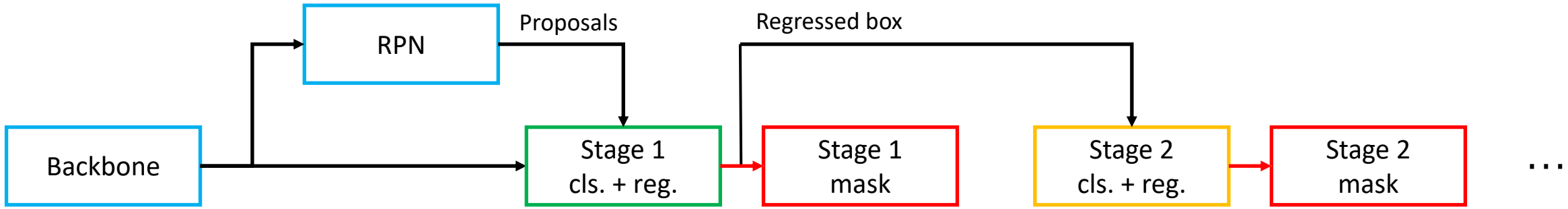
Pipeline

Task cascade: ordinal bbox prediction and mask prediction



Pipeline

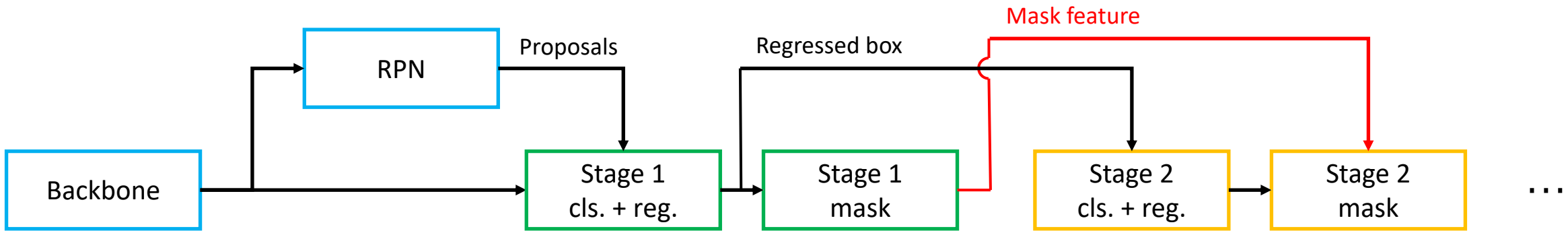
Task cascade: ordinal bbox prediction and mask prediction



Problem: no connection between mask branches of different stages

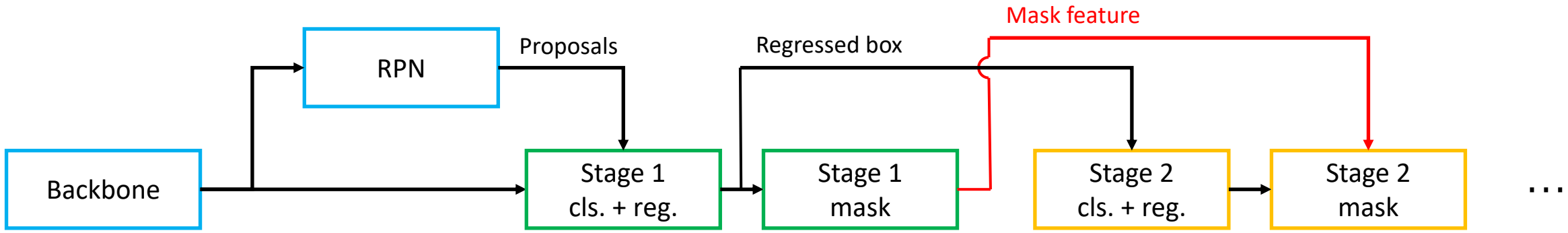
Pipeline

Interleaved execution: box cascade & mask cascade



Pipeline

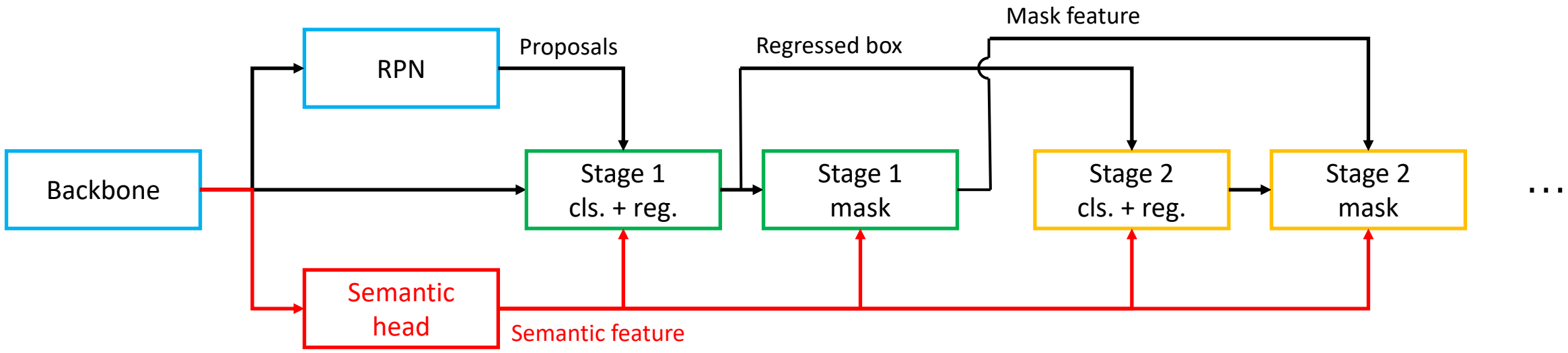
Interleaved execution: box cascade & mask cascade



Problem: contextual information is not much explored

Pipeline

Hybrid branching: additional semantic segmentation branch



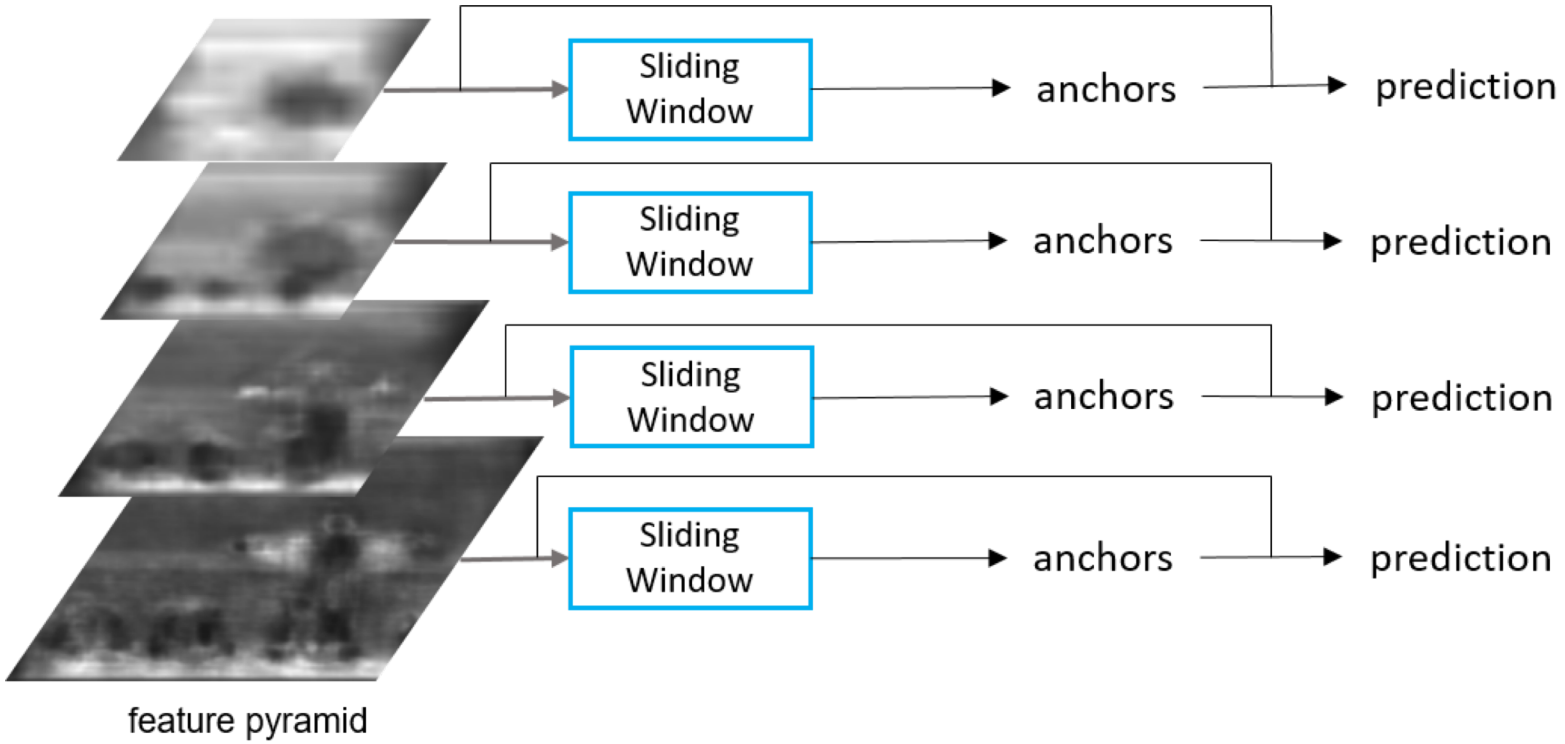
Overview

1. We developed a **hybrid cascading and branching** pipeline for detection and segmentation.
2. We proposed a **feature guided anchoring** scheme to improve the average recall (AR) of RPN by 10 points. (submitted to AAAI 2019)
3. We designed a new backbone **FishNet**. (accepted to NIPS 2018)

Guided Anchoring

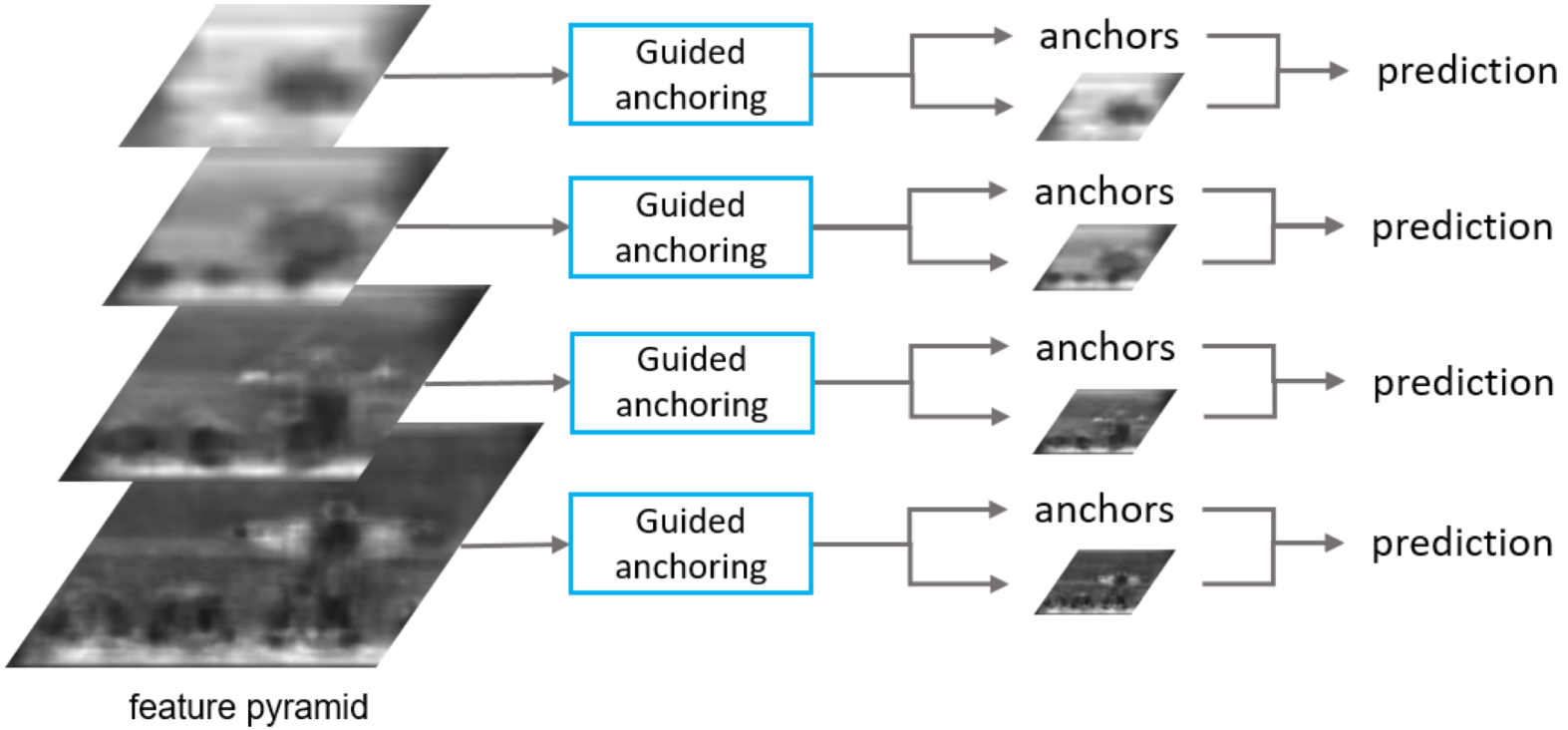
- From sliding window to sparse, non-uniform distribution
- From predefined shapes to learnable, arbitrary shapes
- Refine features based on anchor shapes

Guided Anchoring



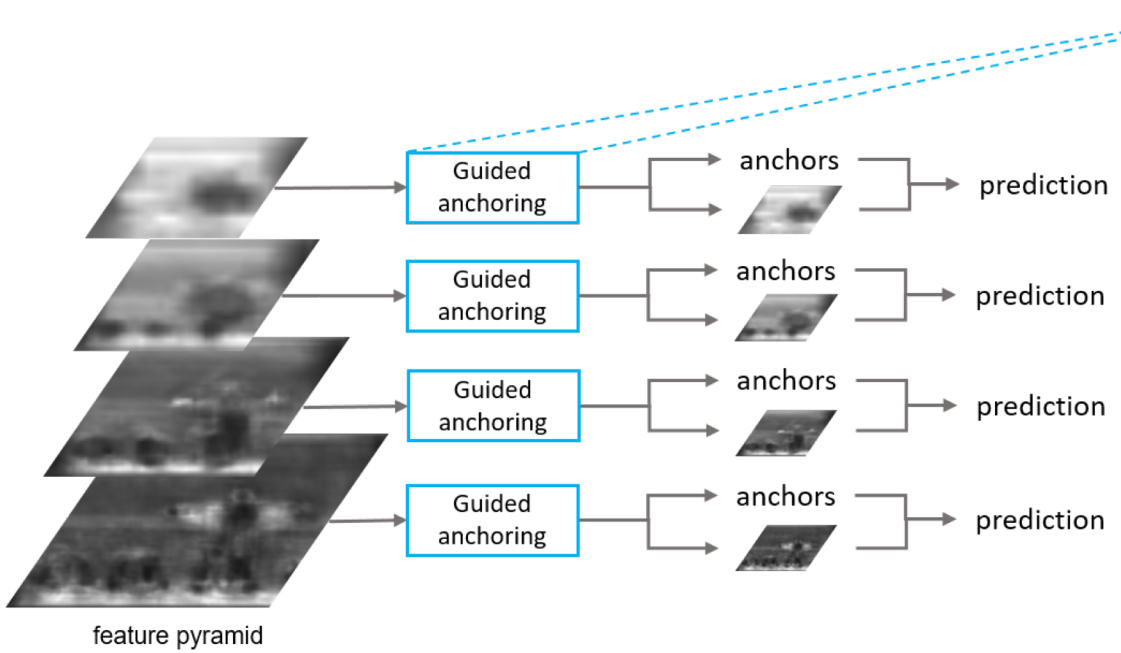
RPN w/ FPN

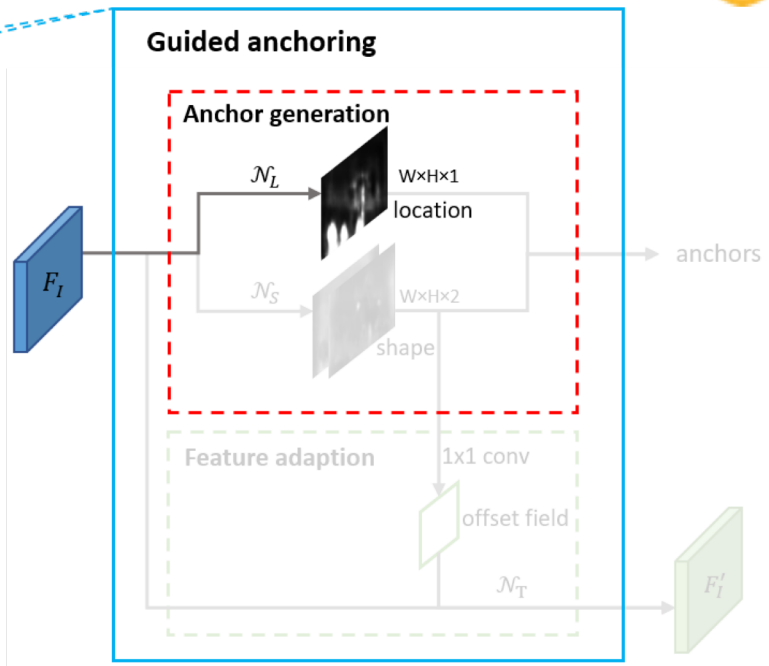
Guided Anchoring

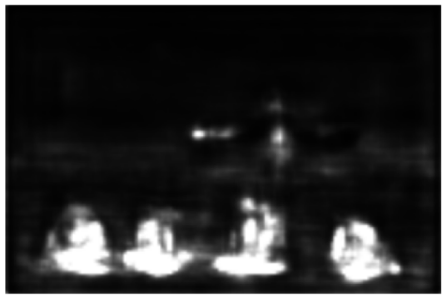


GA-RPN w/ FPN

Guided Anchoring





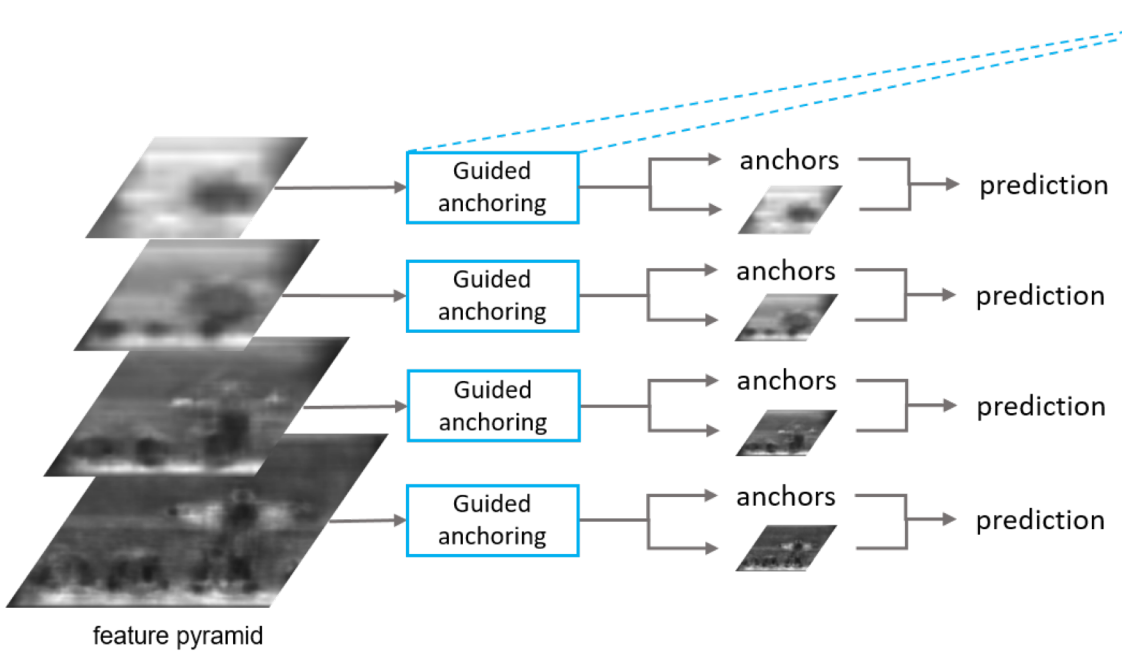


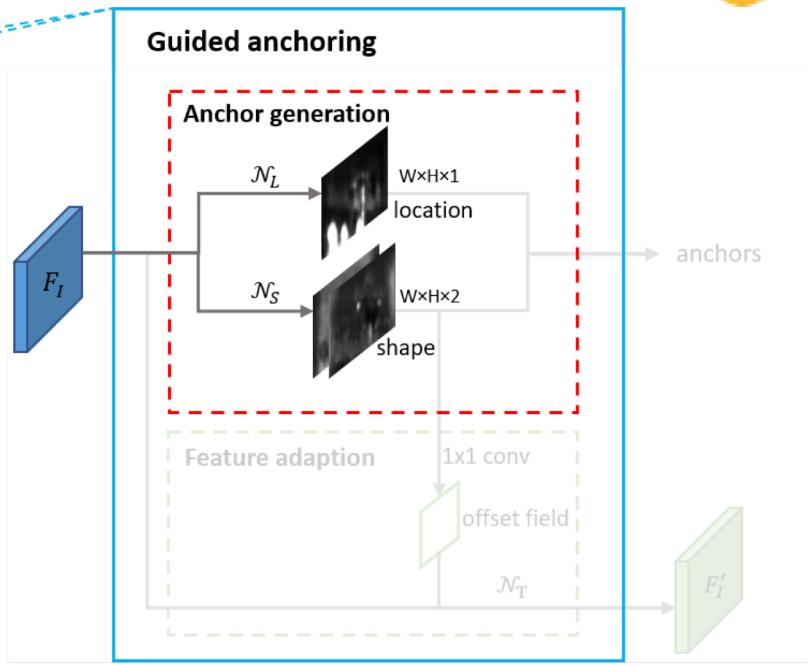
predicted anchor probabilities

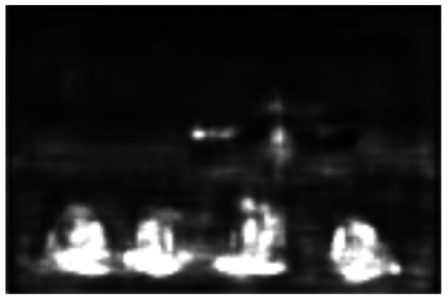
predicted anchor aspect ratios

predicted anchors

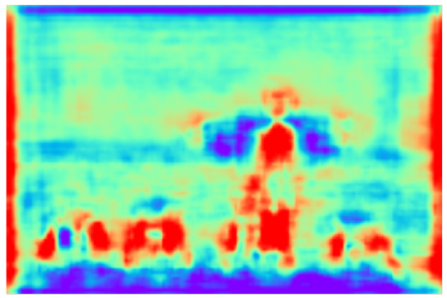
Guided Anchoring



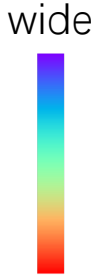




predicted anchor probabilities

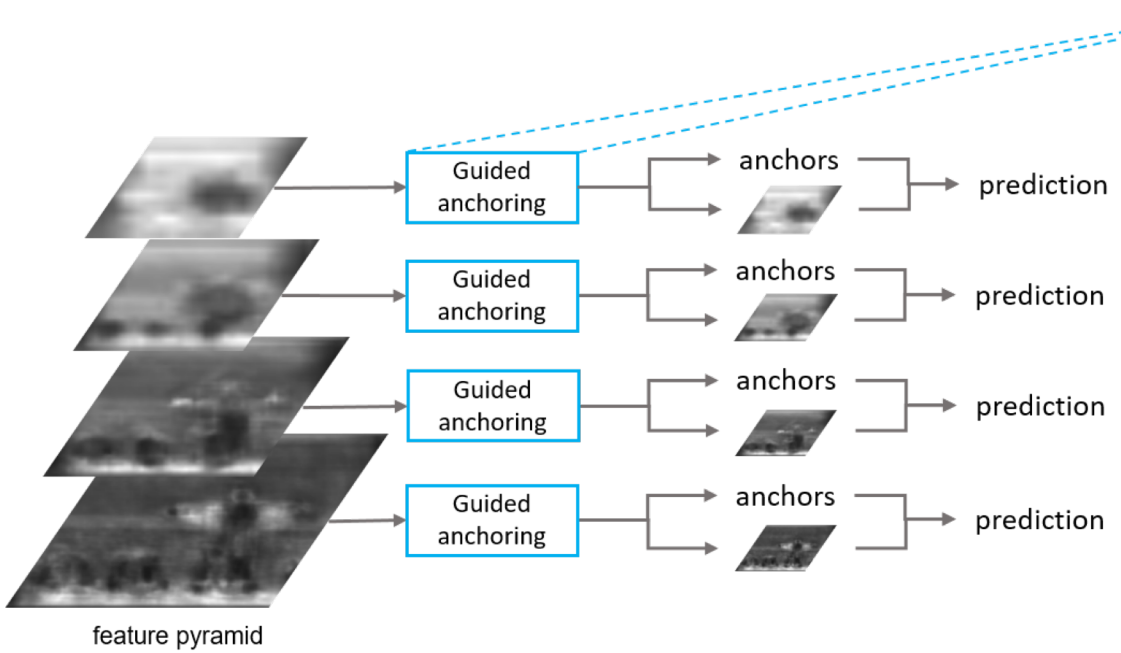


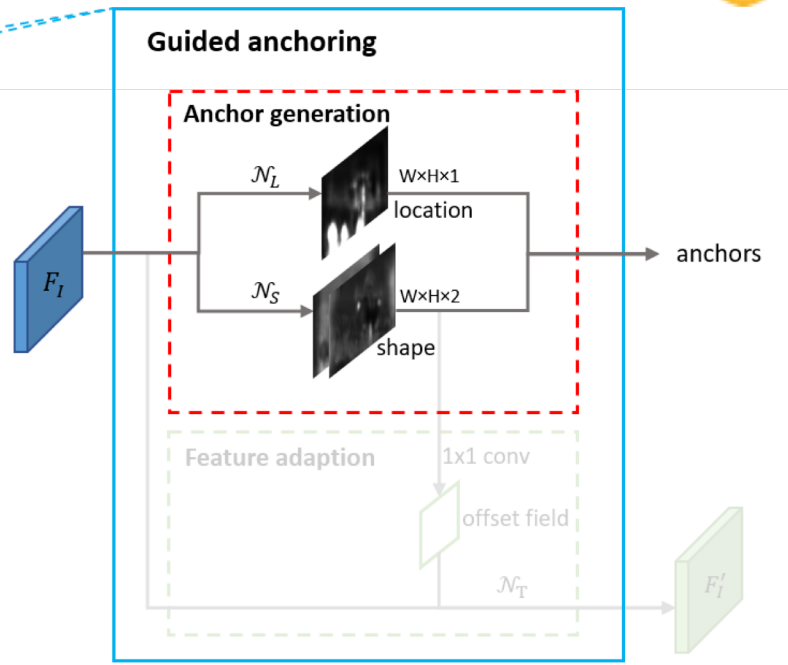
predicted anchor aspect ratios

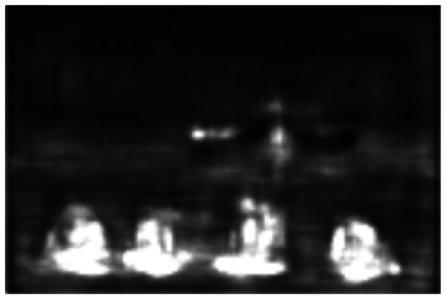


predicted anchors

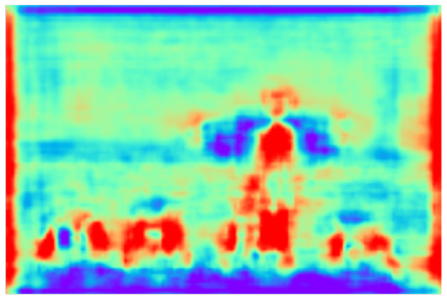
Guided Anchoring



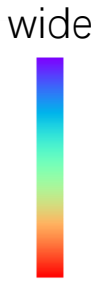




predicted anchor probabilities

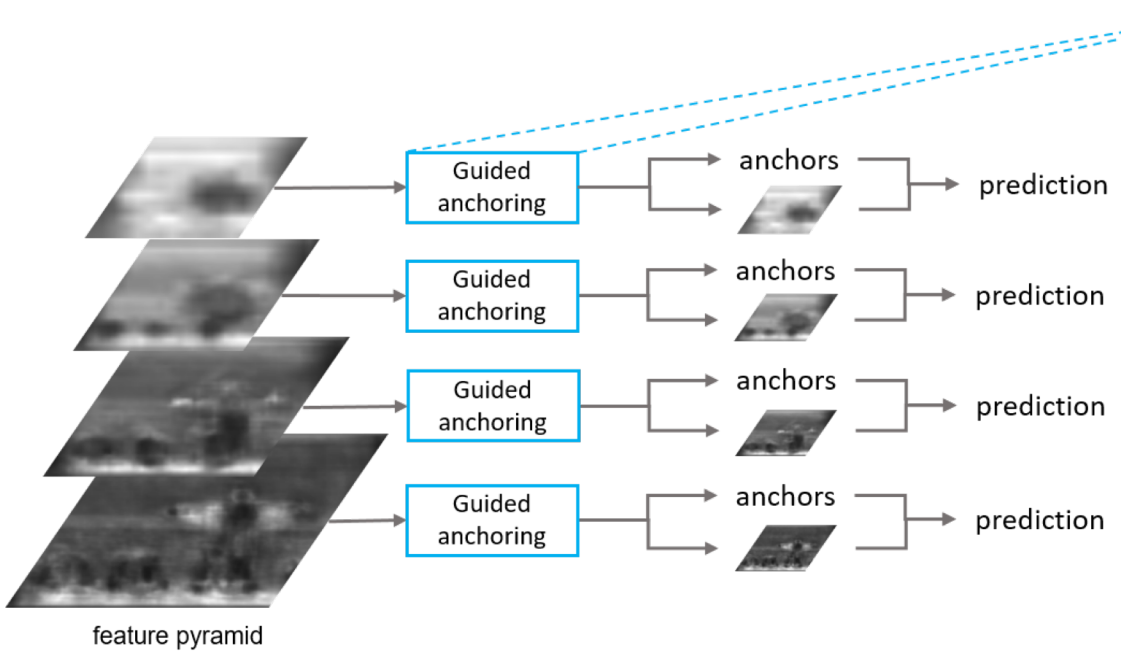


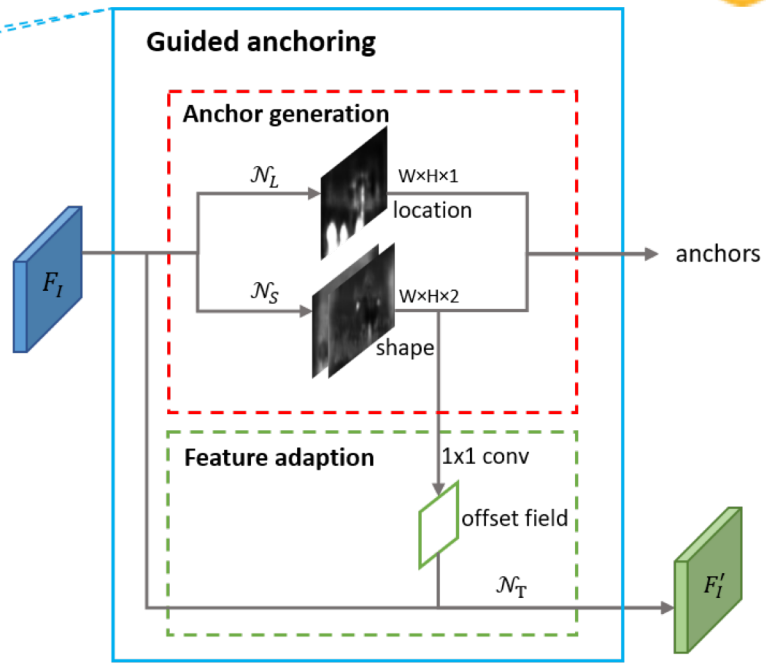
predicted anchor aspect ratios

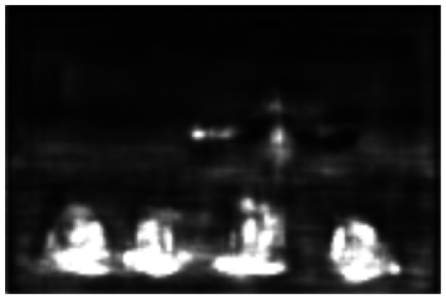


predicted anchors

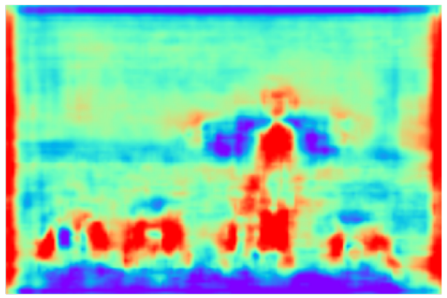
Guided Anchoring



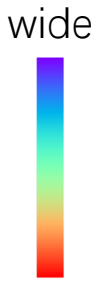




predicted anchor probabilities

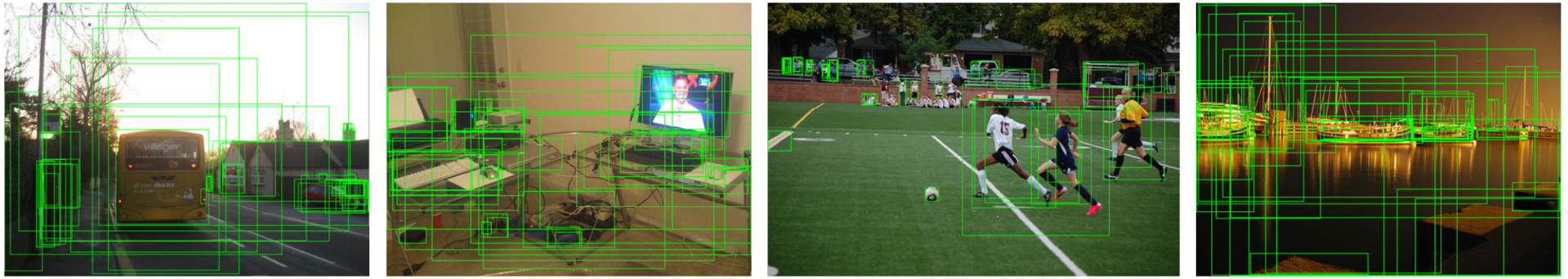


predicted anchor aspect ratios



predicted anchors

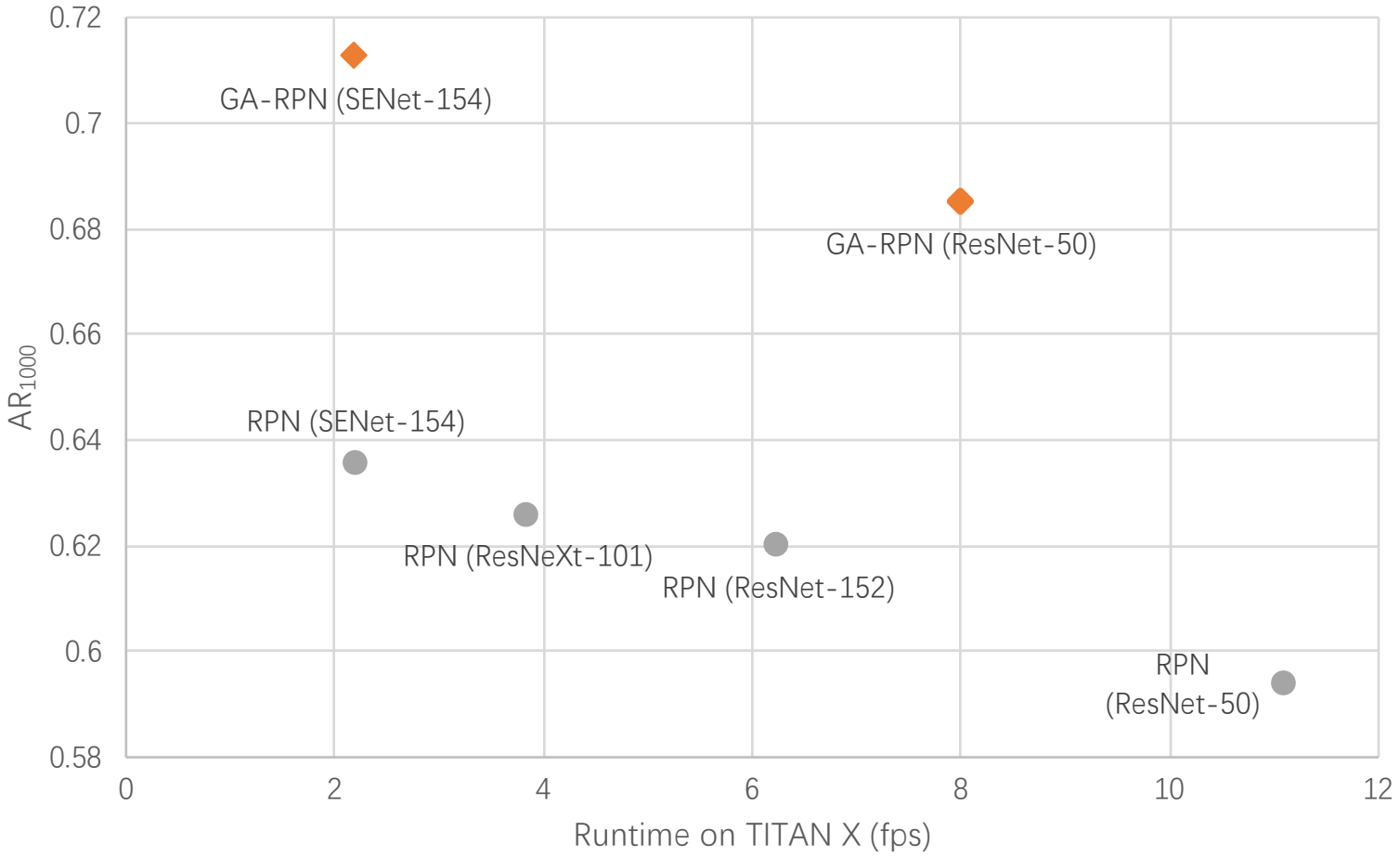
Guided Anchoring



RPN

GA-RPN

Guided Anchoring



Overview

1. We developed a **hybrid cascading and branching** pipeline for detection and segmentation.
2. We proposed a **feature guided anchoring** scheme to improve the average recall (AR) of RPN by 10 points. (submitted to AAAI 2019)
3. We designed a new backbone **FishNet**. (accepted to NIPS 2018)

FishNet

Motivation

- The basic principles for designing CNN for region and pixel level tasks are **diverging** from the principles for image classification.
- Unify the advantages of networks designed for region and pixel level tasks in obtaining **deep** features with **high-resolution**.

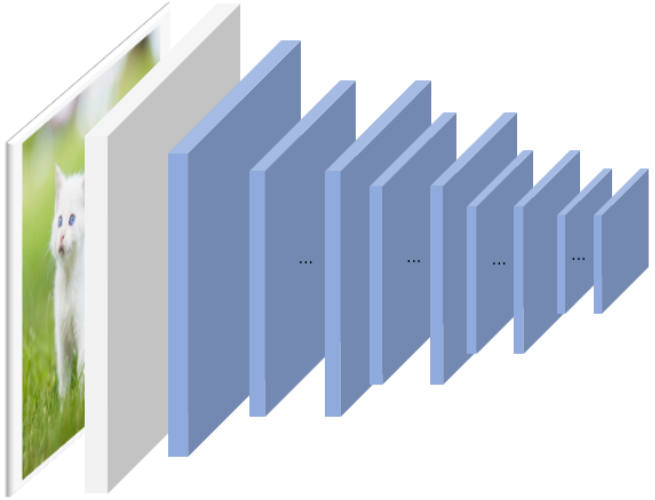
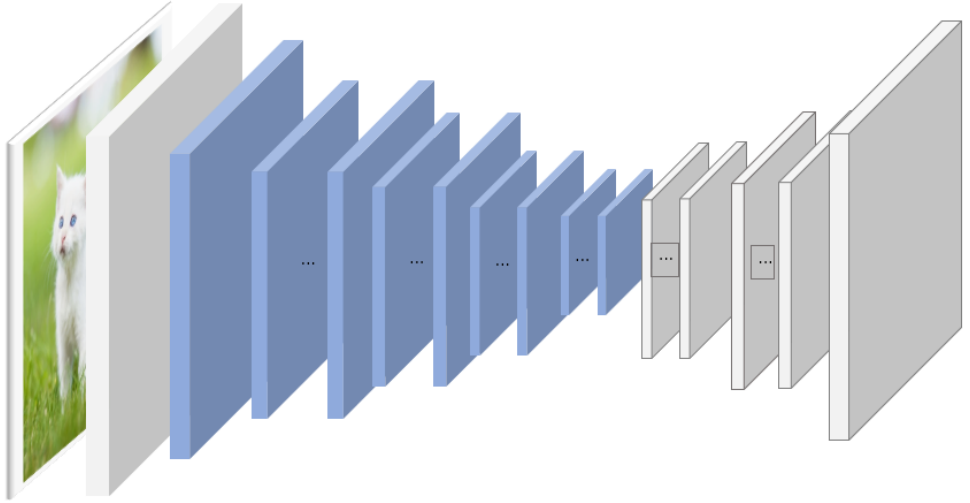


Image classification



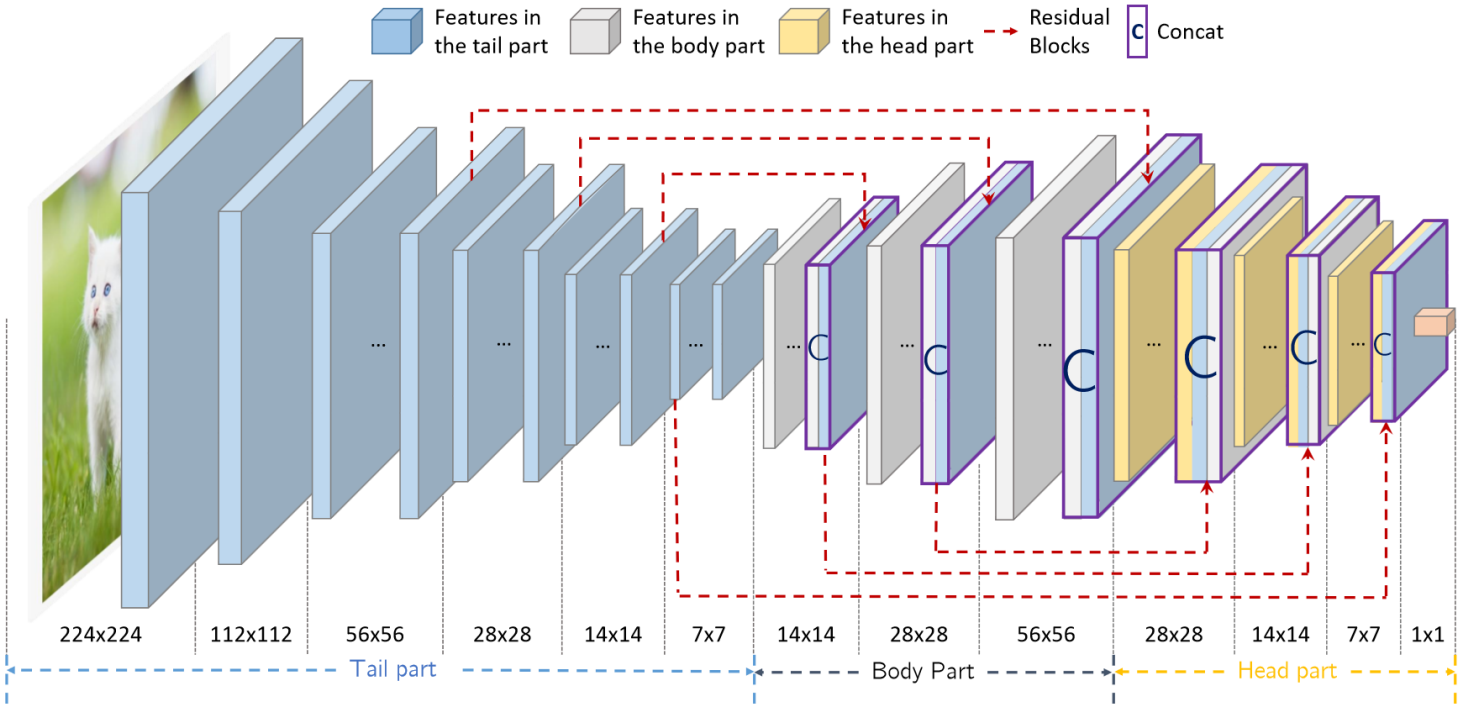
Region and pixel level tasks

Segmentation, pose estimation, detection ...

FishNet

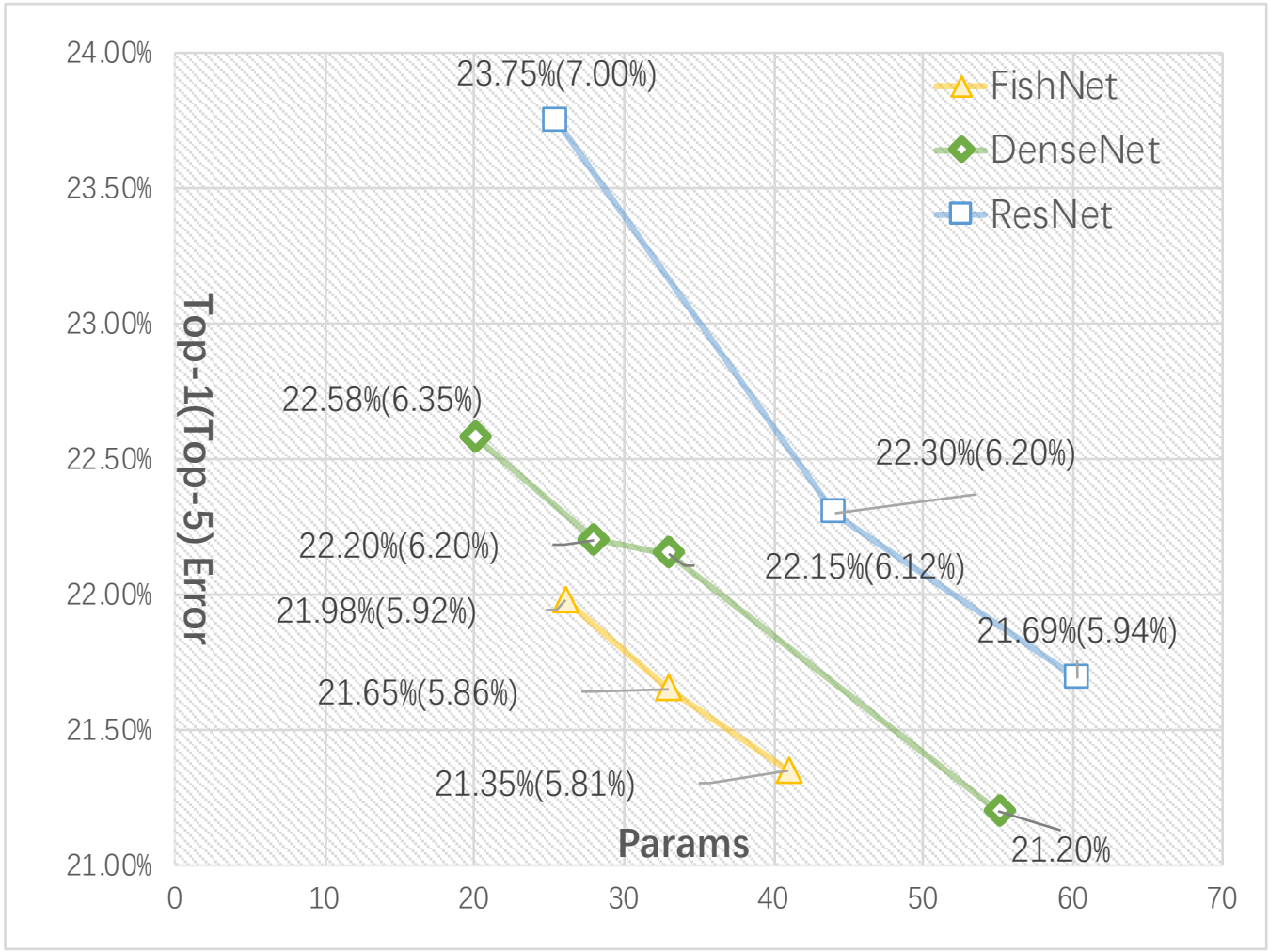
Motivation

- Traditional consecutive down-sampling will prevent the very shallow layers to be directly connected till the end, which may exacerbate the **vanishing gradient problem.**
- Features from varying depths could be used for **refining** each other.



FishNet: A Versatile Backbone for Image, Region, and Pixel Level Prediction, NIPS 2018, accepted.

FishNet



Top-1 Classification Error on ImageNet

FishNet

MS COCO *val-2017* detection and instance segmentation results.

	Instance Segmentation	Object Detection
Backbone	$AP^s/AP^s_S/AP^s_M/AP^s_L$	$AP^d/AP^d_S/AP^d_M/AP^d_L$
ResNet-50 [3]	34.5/15.6/37.1/52.1	38.6/22.2/41.5/50.8
ResNet-50†	34.7/18.5/37.4/47.7	38.7/22.3/42.0/51.2
ResNeXt-50 (32x4d)†	35.7/19.1/38.5/48.5	40.0/23.1/43.0/52.8
FishNet-188	**37.0**/19.8/40.2/50.3	**41.5**/24.1/44.9/55.0
vs. ResNet-50†	**+2.3/+1.3/+2.8/+2.6**	**+2.8/+1.8/+2.9/+3.8**
vs. ResNeXt-50†	**+1.3/+0.7/+1.7/+1.8**	**+1.5/+1.0/+1.9/+2.2**

Experiments

Training/Testing details

1. Training scales
   - short edge: random sampled from 400 ~ 1400
   - long edge: 1600
2. Test scales
   - (600, 900), (800, 1200), (1000, 1500), (1200, 1800), (1400, 2100)
3. Pipeline
   - Joint training
   - Finetune with GA-RPN proposals
   - Test with GA-RPN proposals
4. Resources
   - 32 Tesla V100 GPUs (16GB) for 3 days

Experiments

Backbones

- SENet-154
- ResNeXt101 (64*4d)
- ResNeXt101 (32*8d)
- DPN-107
- FishNet

~0.8 points higher

comparable

Experiments

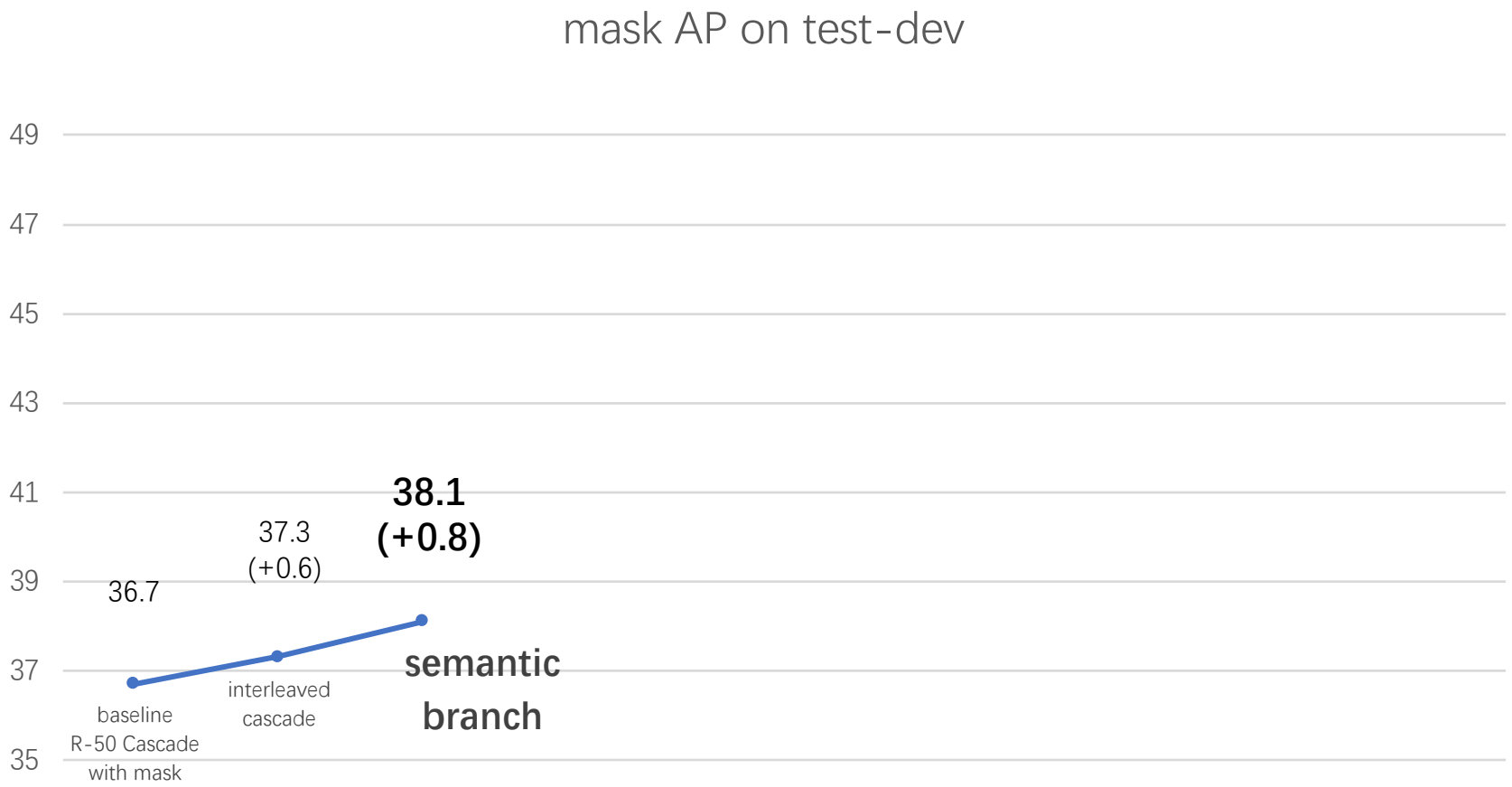
Other tricks

- w/ SoftNMS
- w/o OHEM
- w/o classwise balance sampling
- w/o voting for bbox or mask

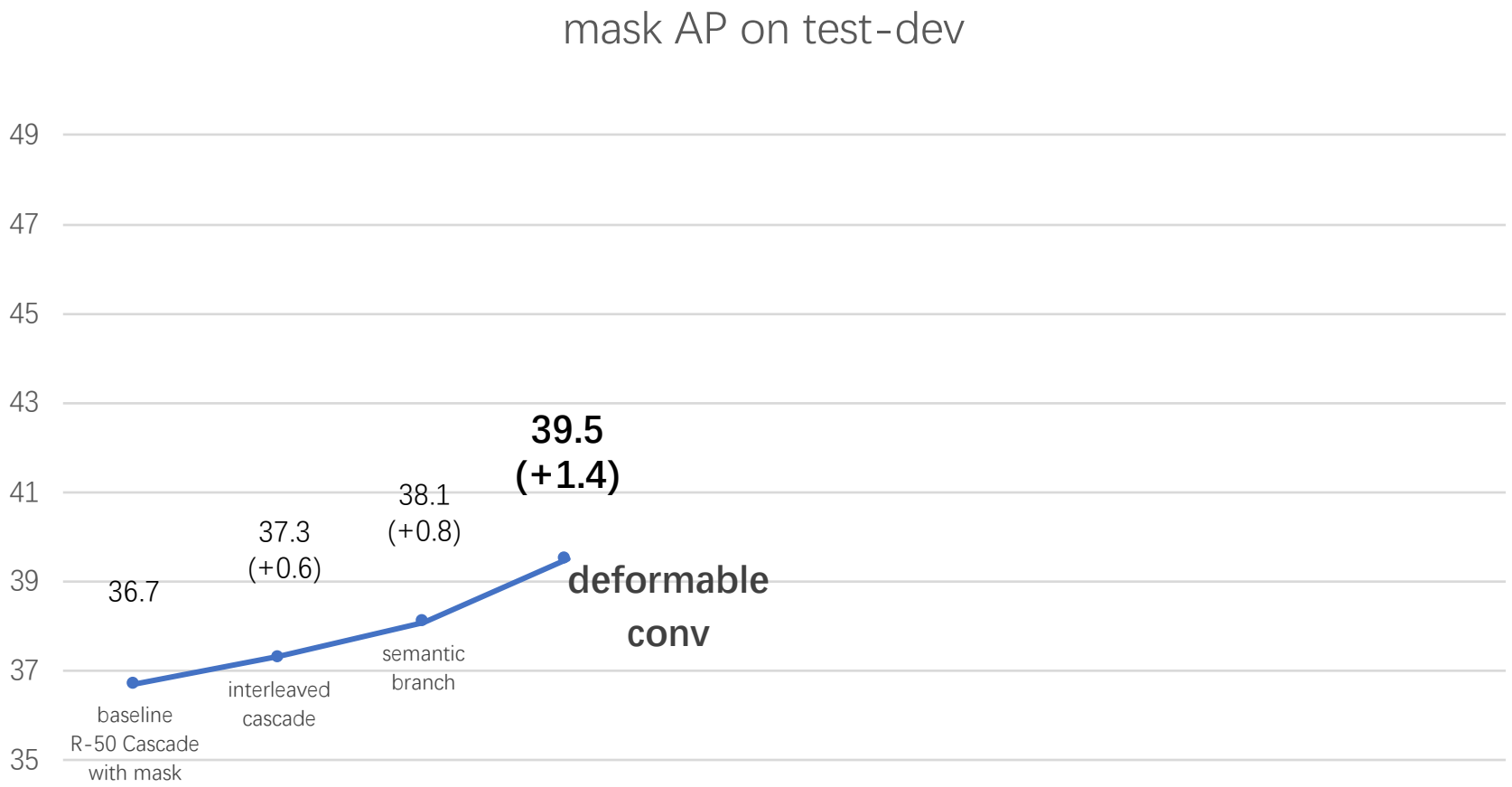
Experiments

Experiments

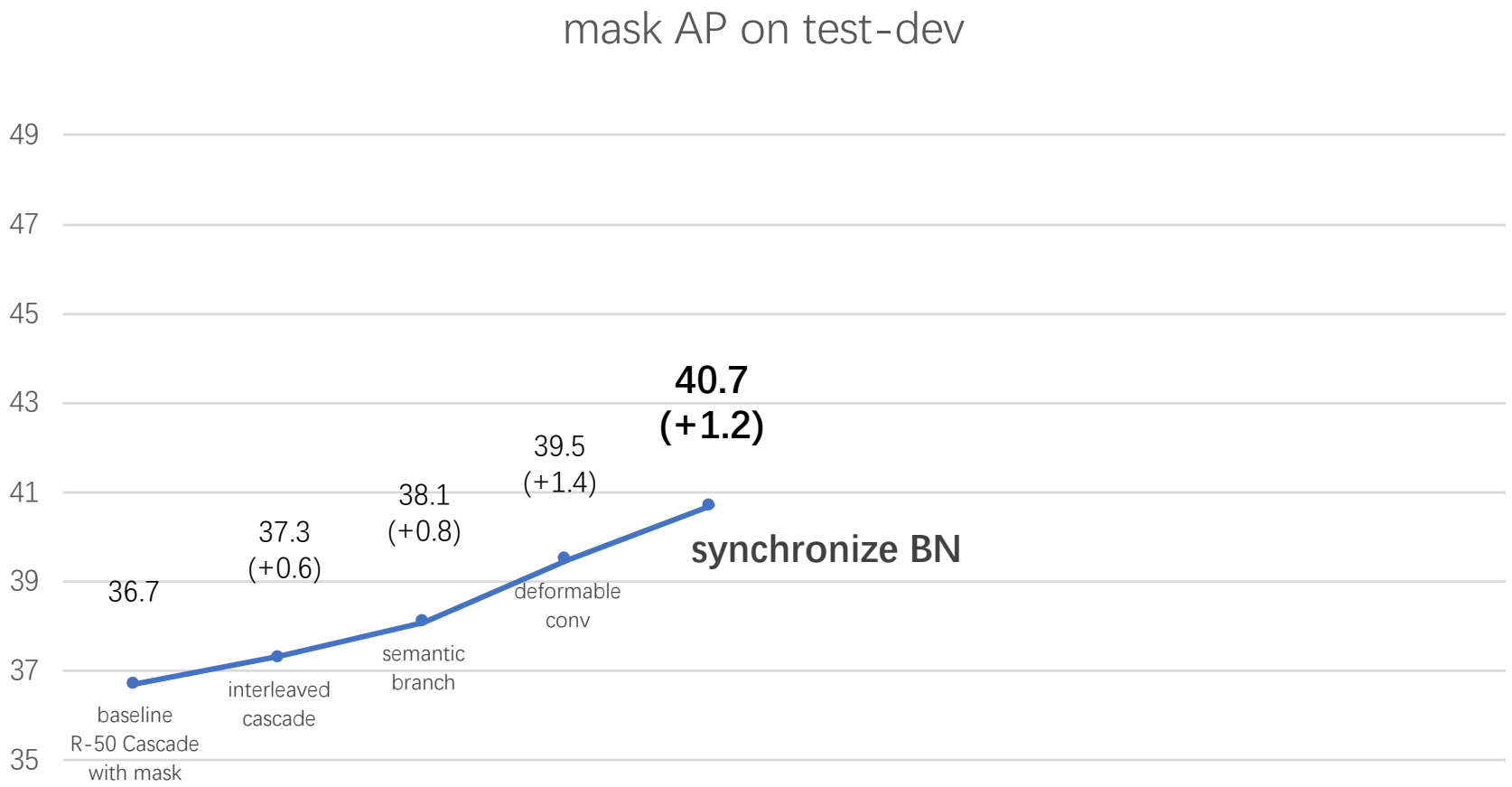
Experiments



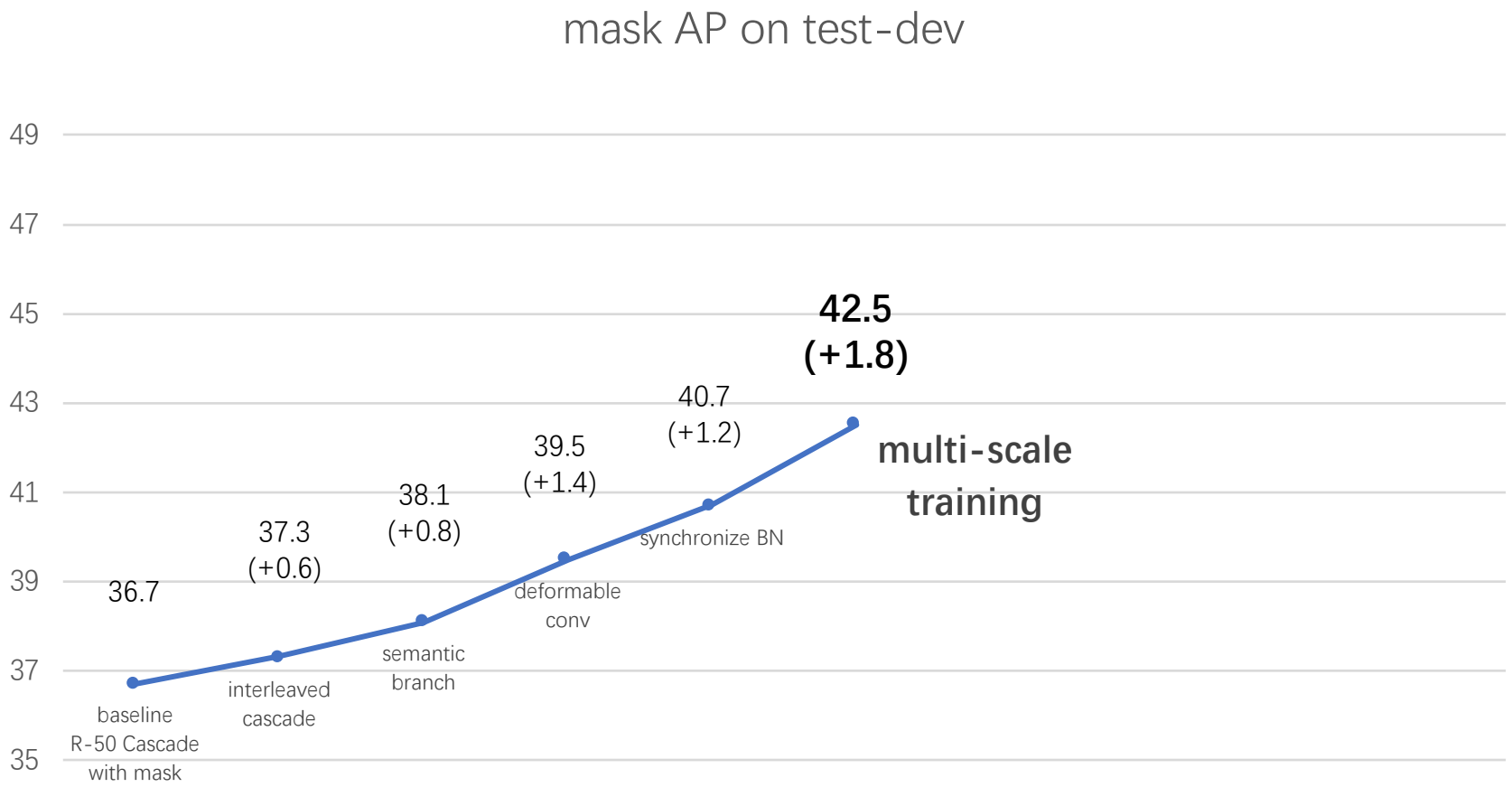
Experiments



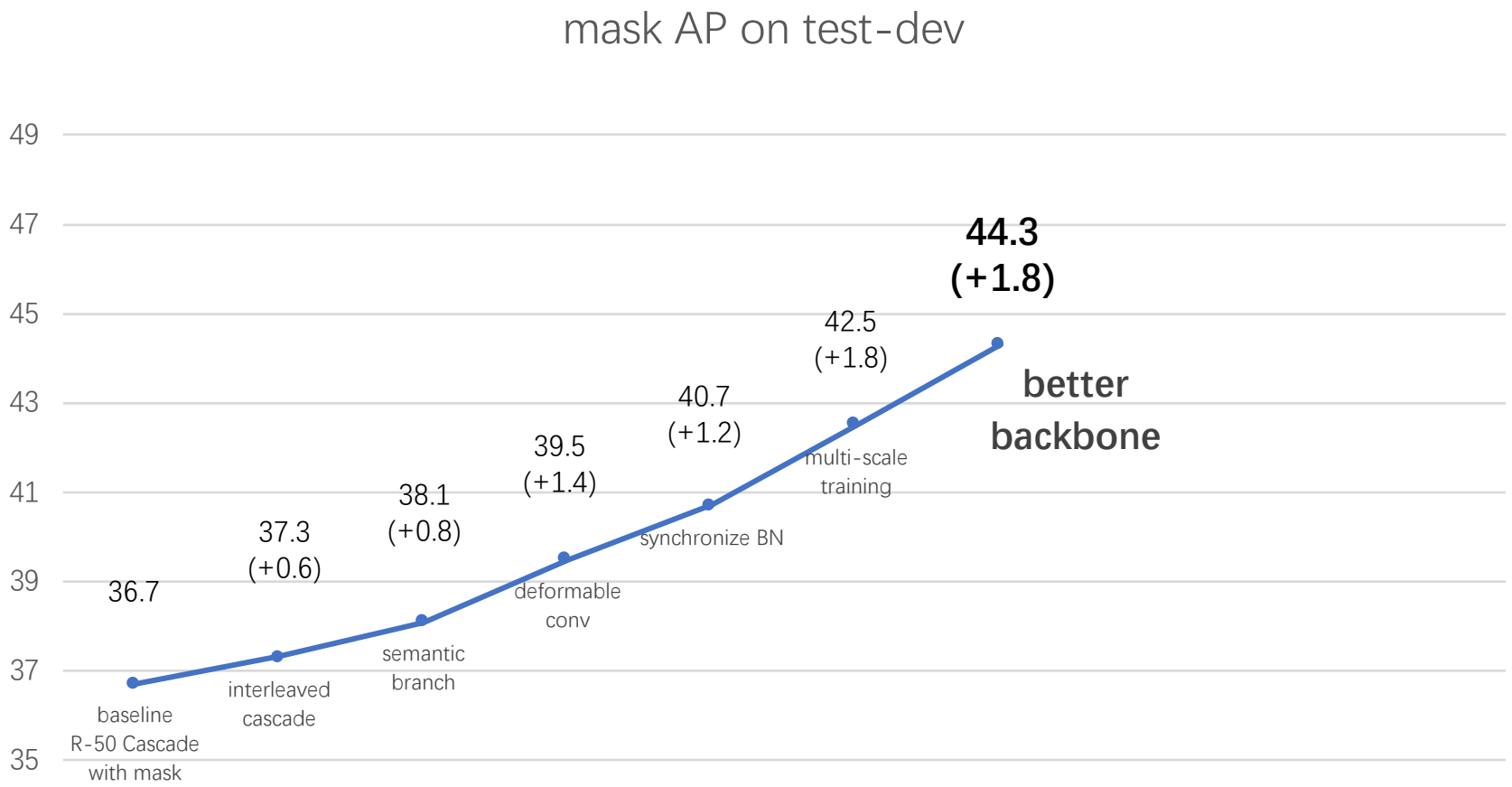
Experiments



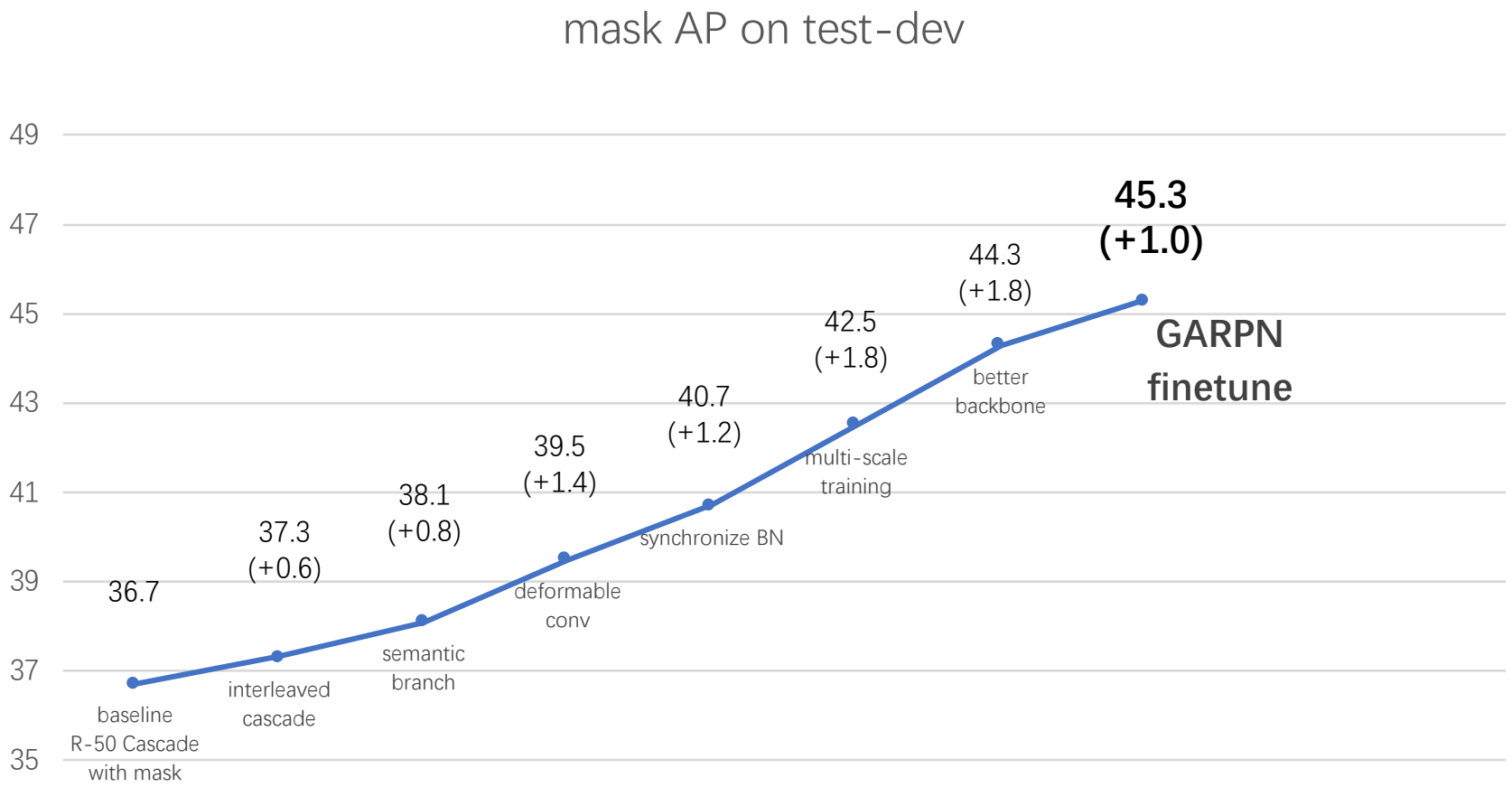
Experiments



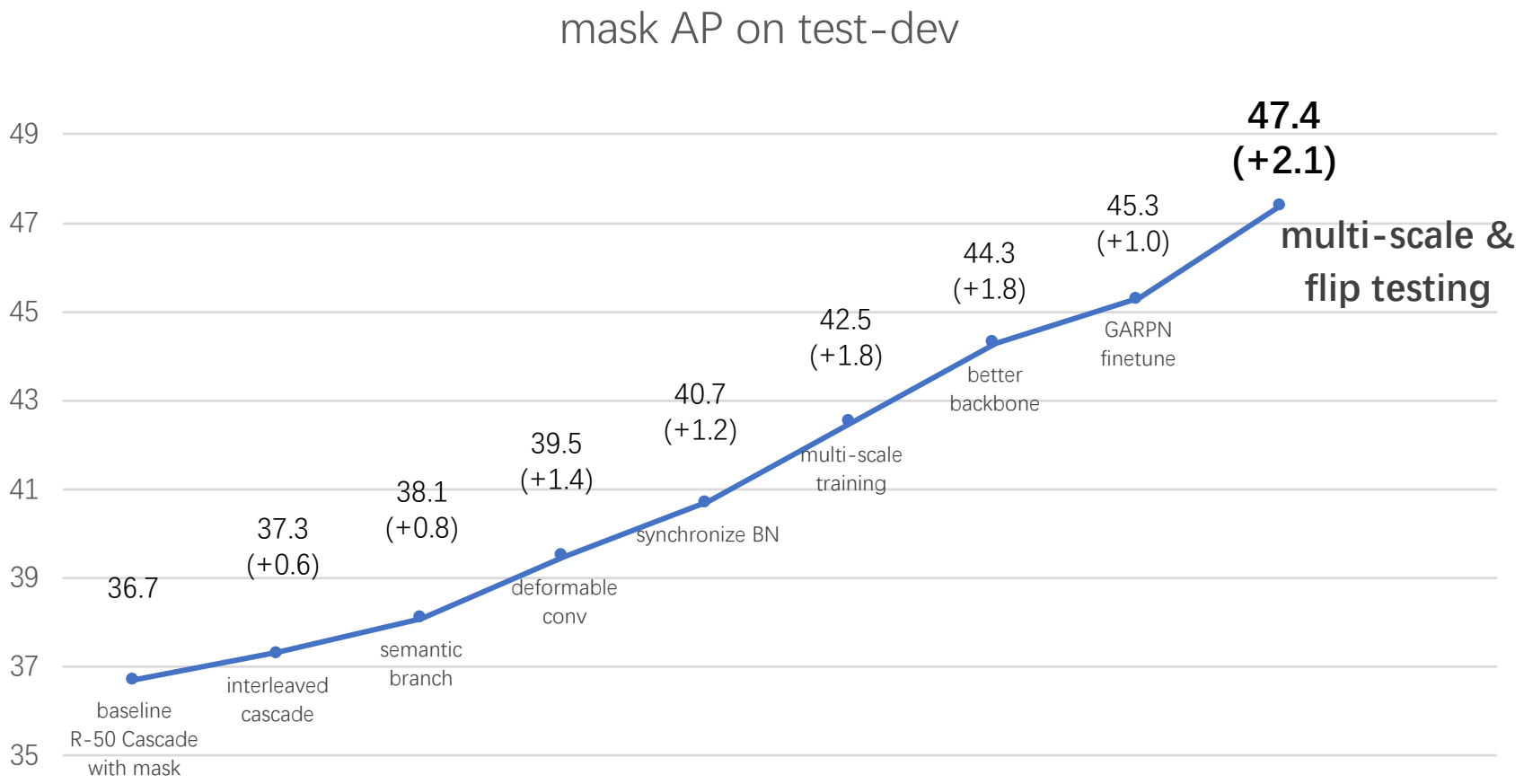
Experiments



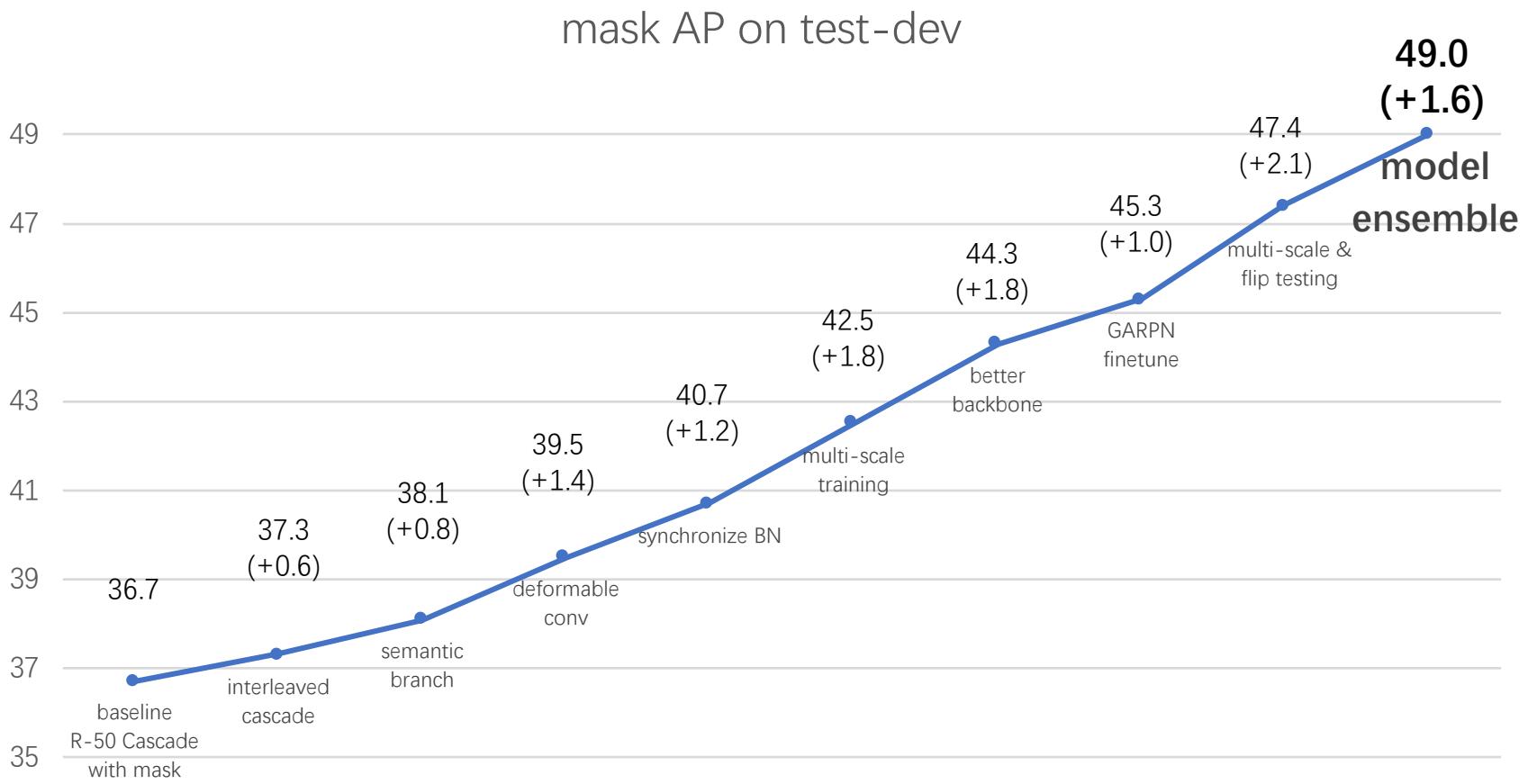
Experiments



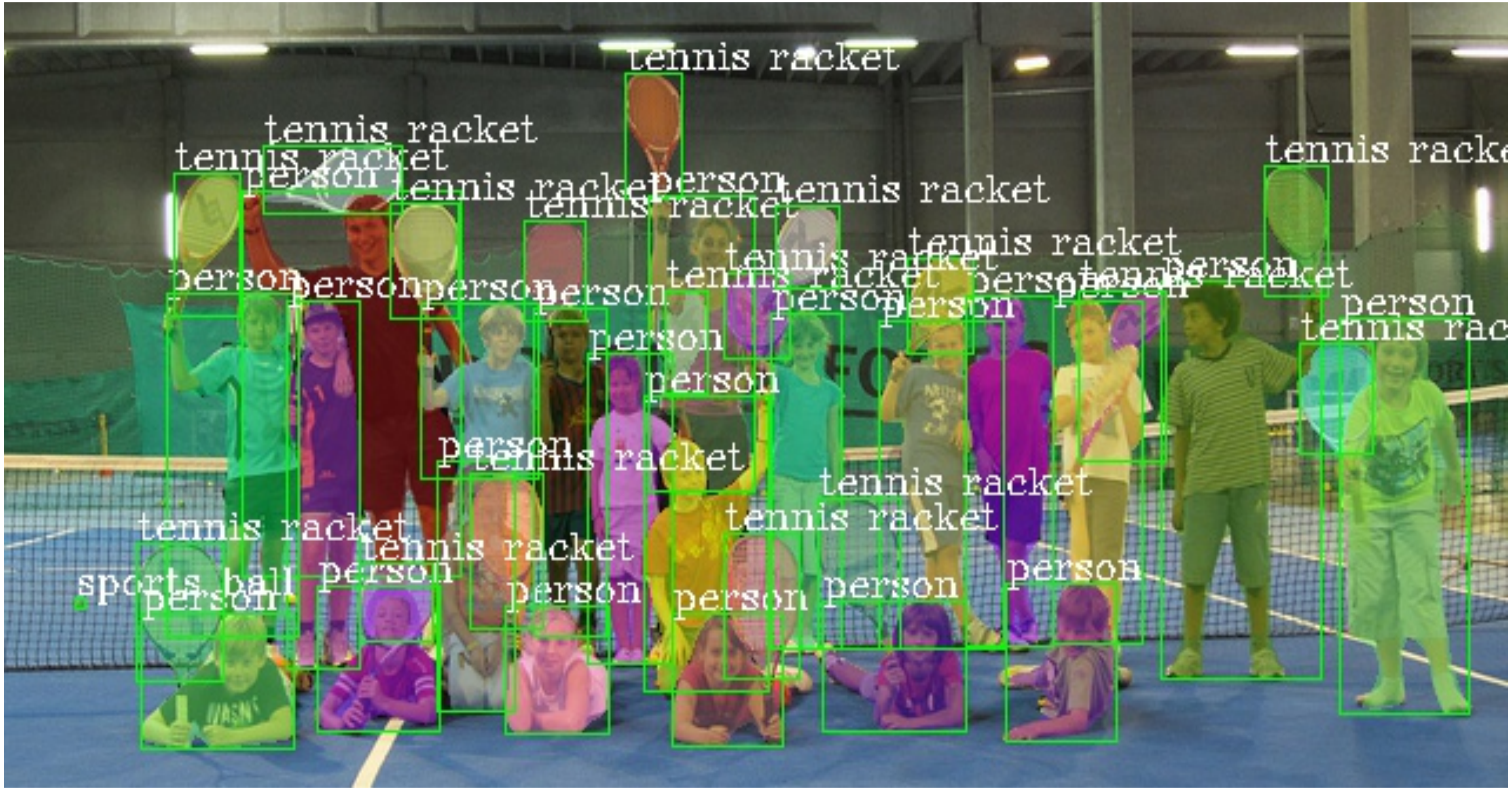
Experiments



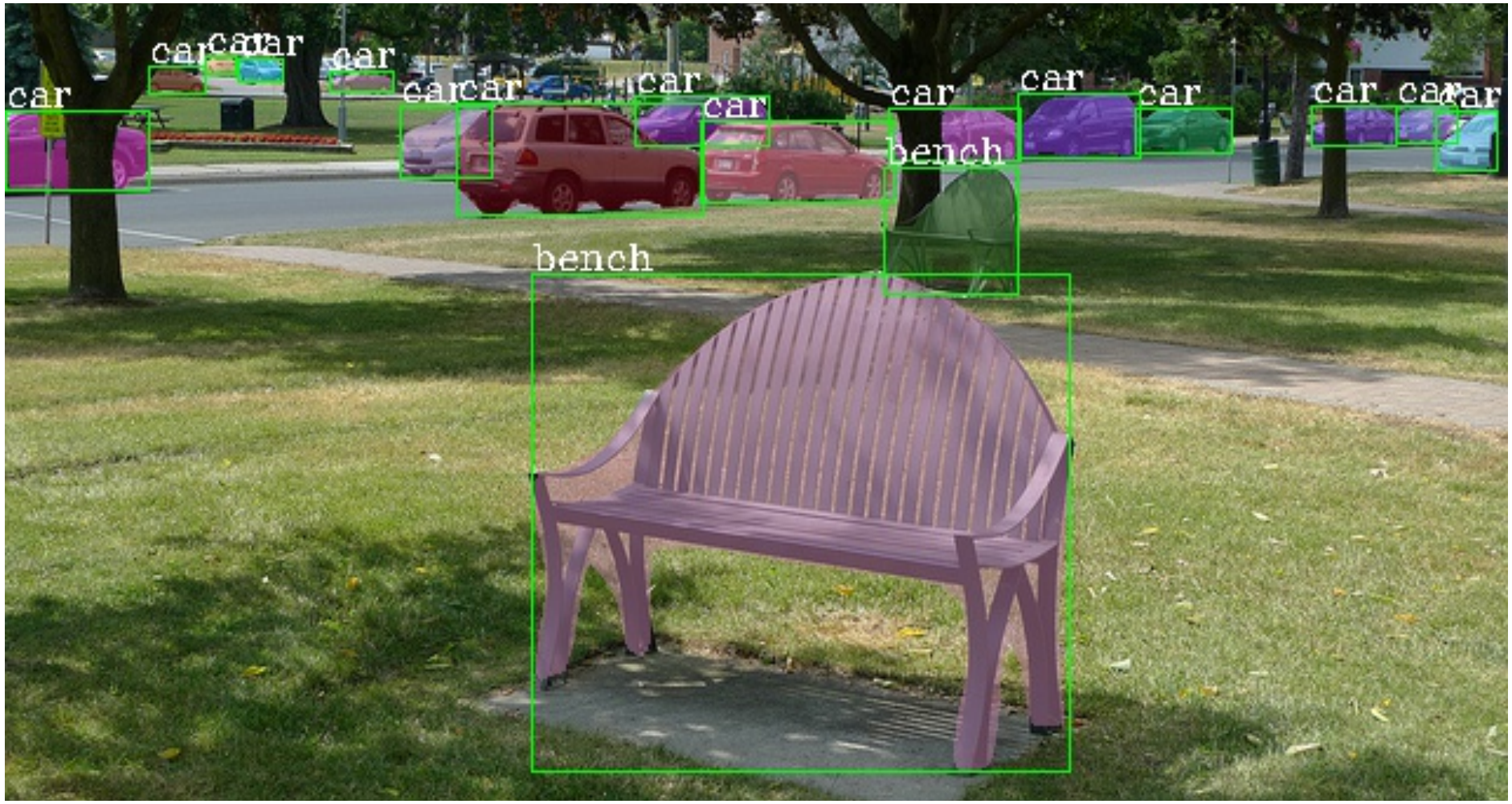
Experiments



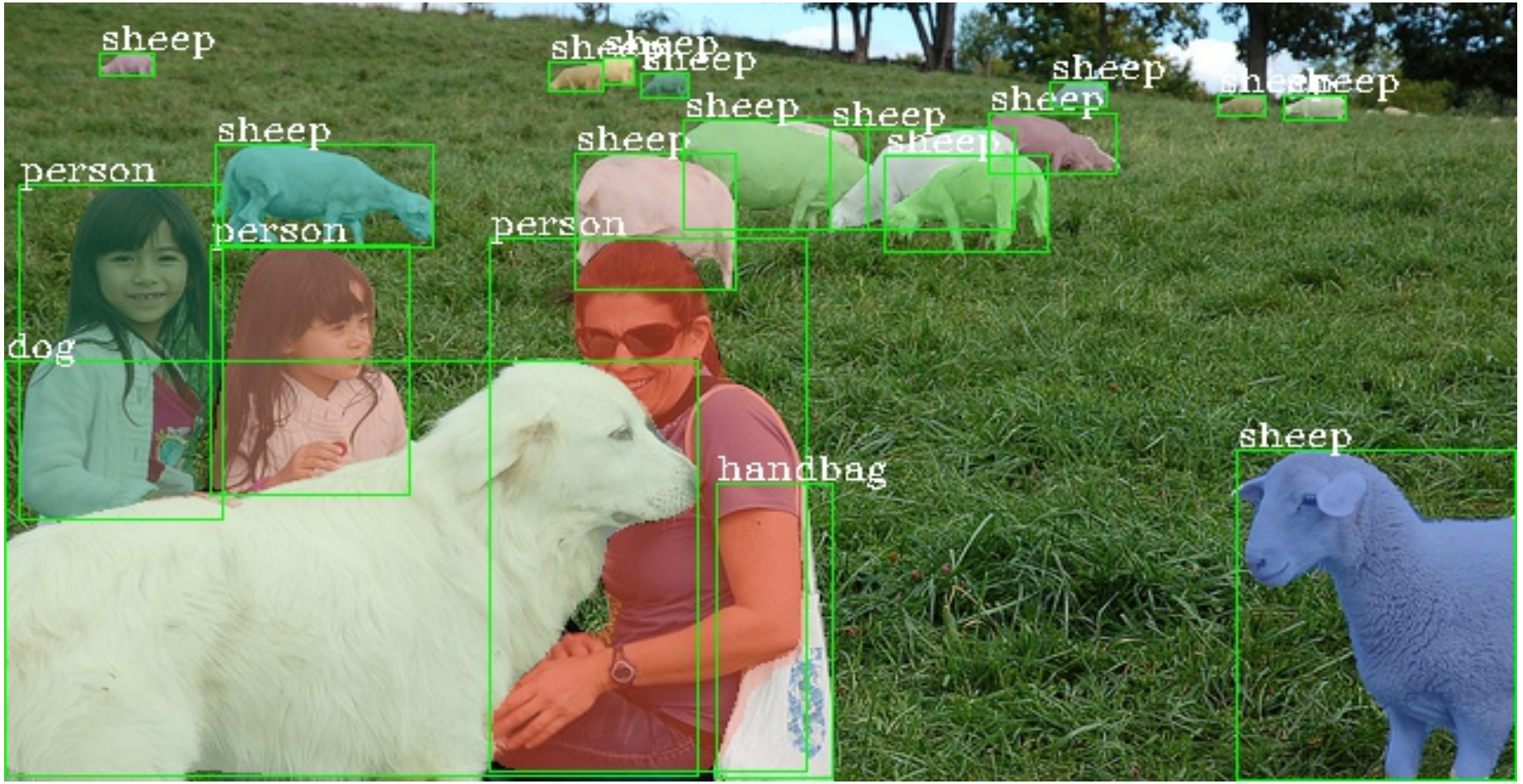
Visualization



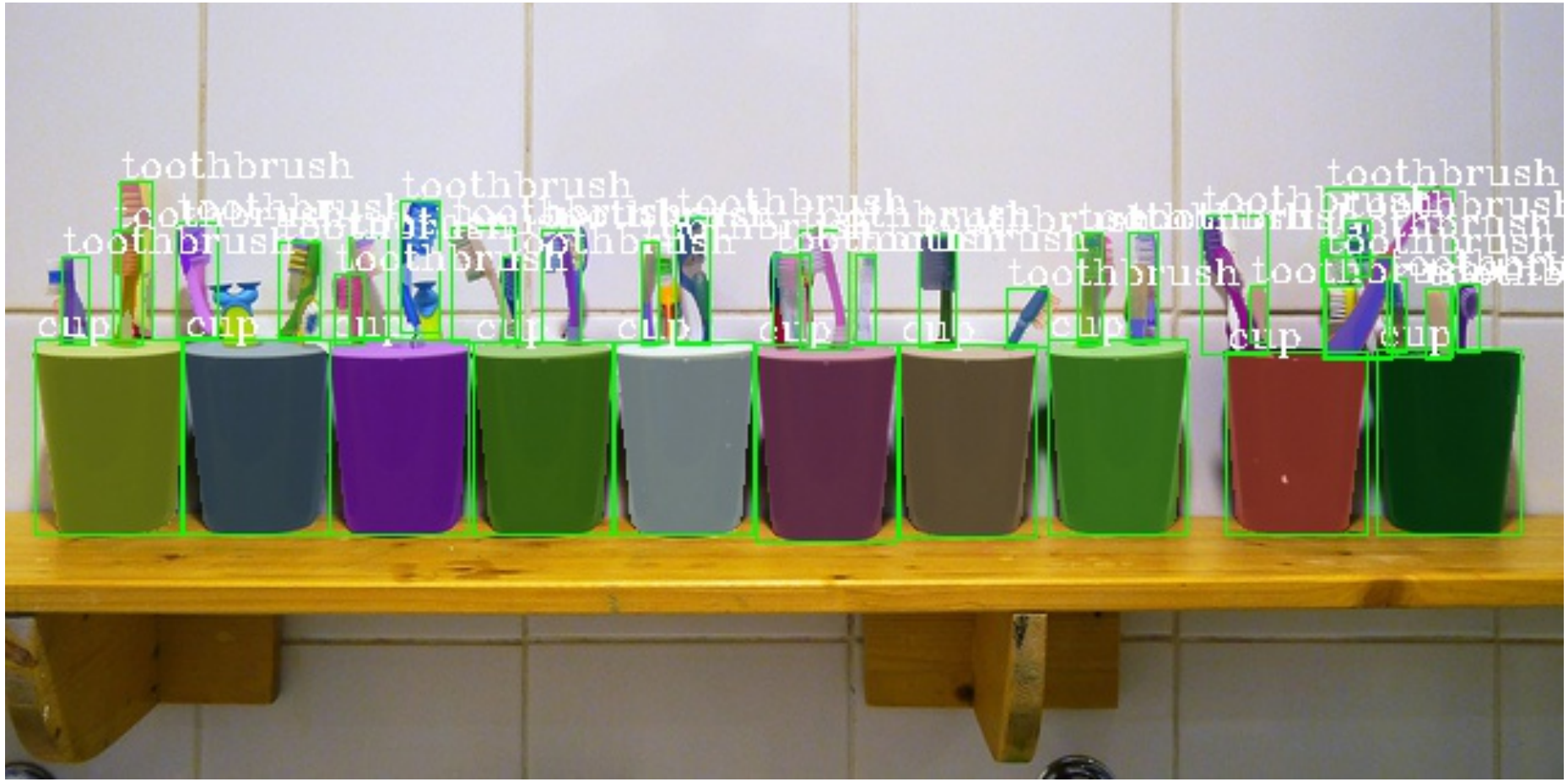
Visualization



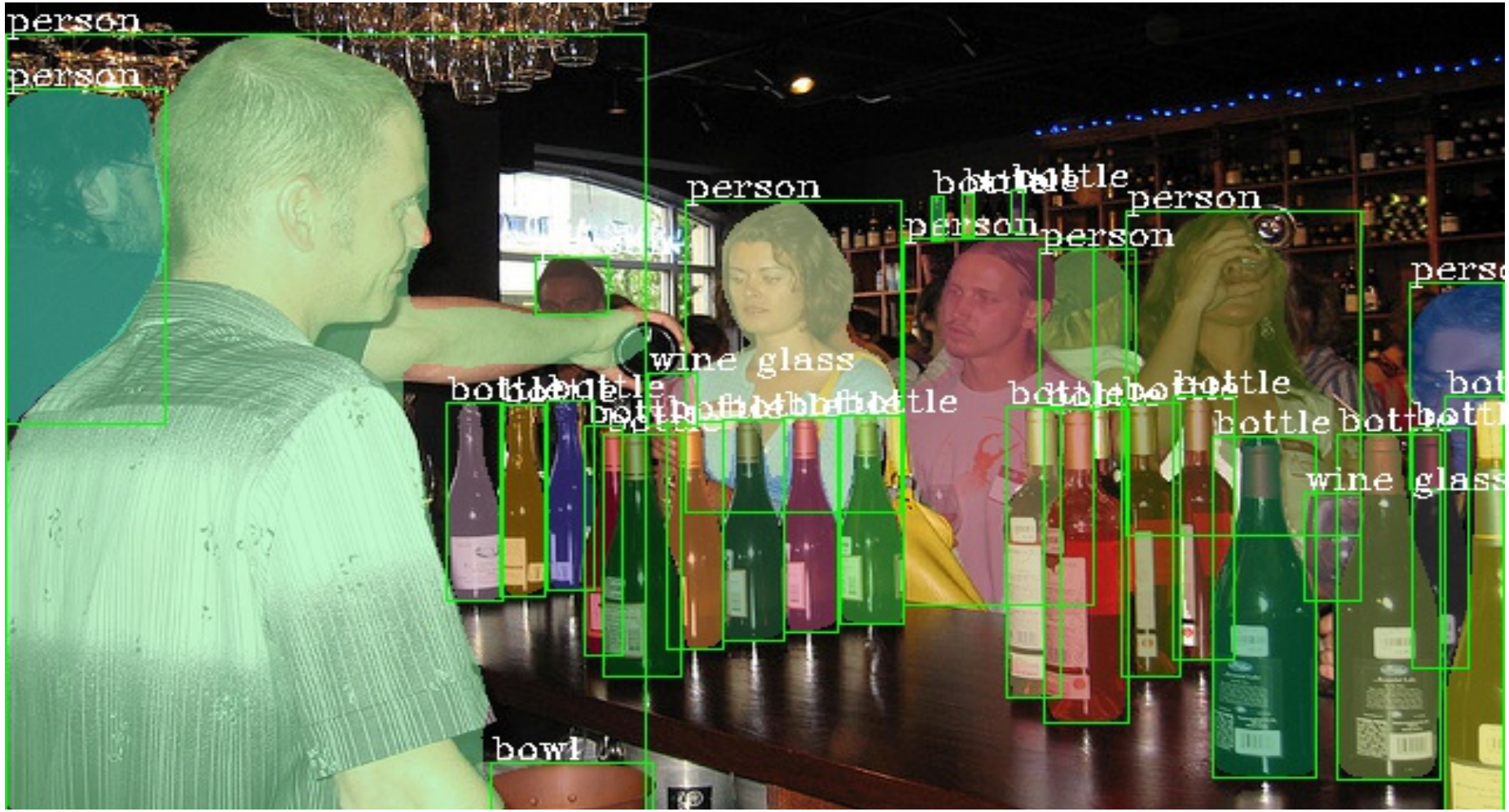
Visualization



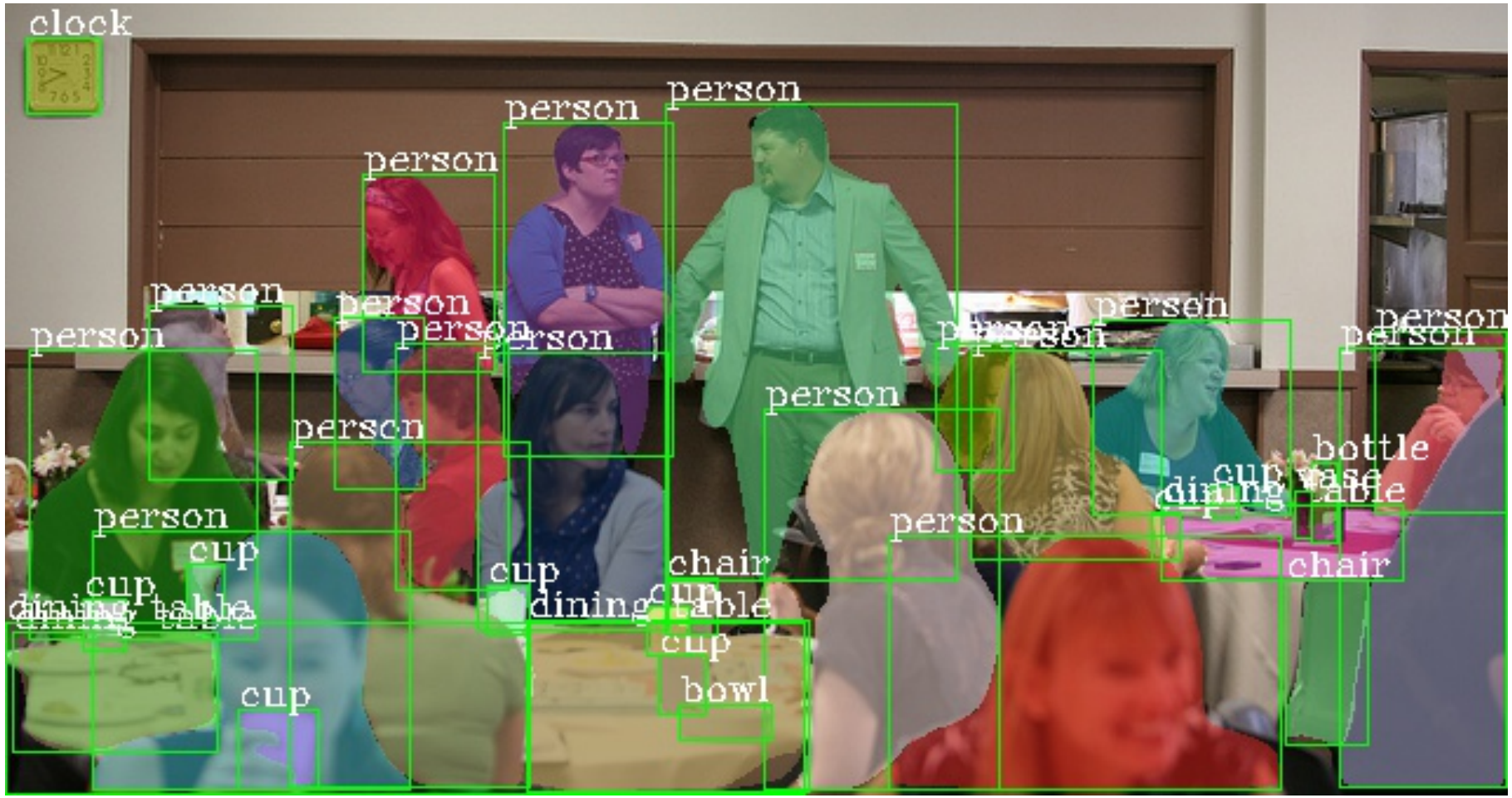
Visualization



Visualization



Visualization



Experience

1. What can bring large gains?

Fundamental improvements of pipelines and structures

- Mask R-CNN
- FPN
- Cascade R-CNN
- (Synchronized) BN
- Deformable ConvNet
- …

Experience

2. What may not?

Improvements of specific modules

- Precise RoI Pooling
- DetNet
- GCN
- Fitness NMS

Extra marginal components

- ASPP
- Spatial attention
- Additional R-CNN/PSPNet

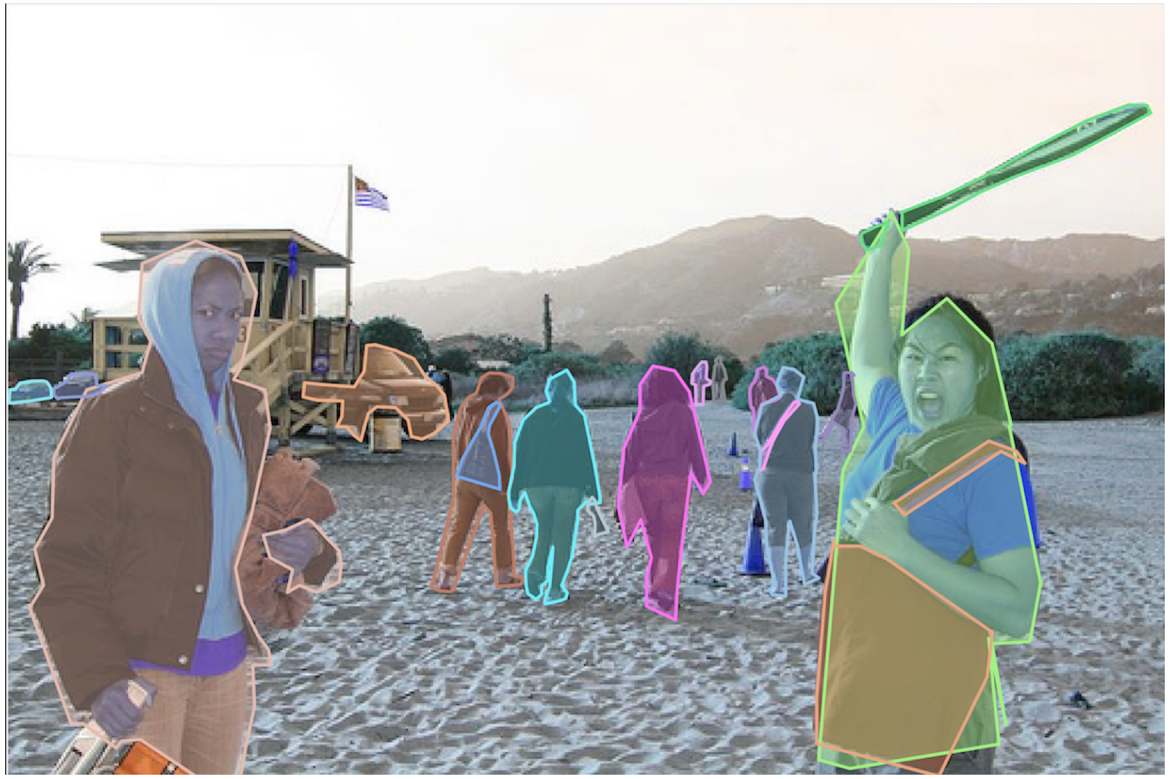
Experience

2. What may not?

- Increasing model complexity can eat most of the gains
- Combination of ideas is not trivial
- May not be universal or robust
- Time is limited or wrong implementation

Experience

3. The annotation quality may limit the performance.

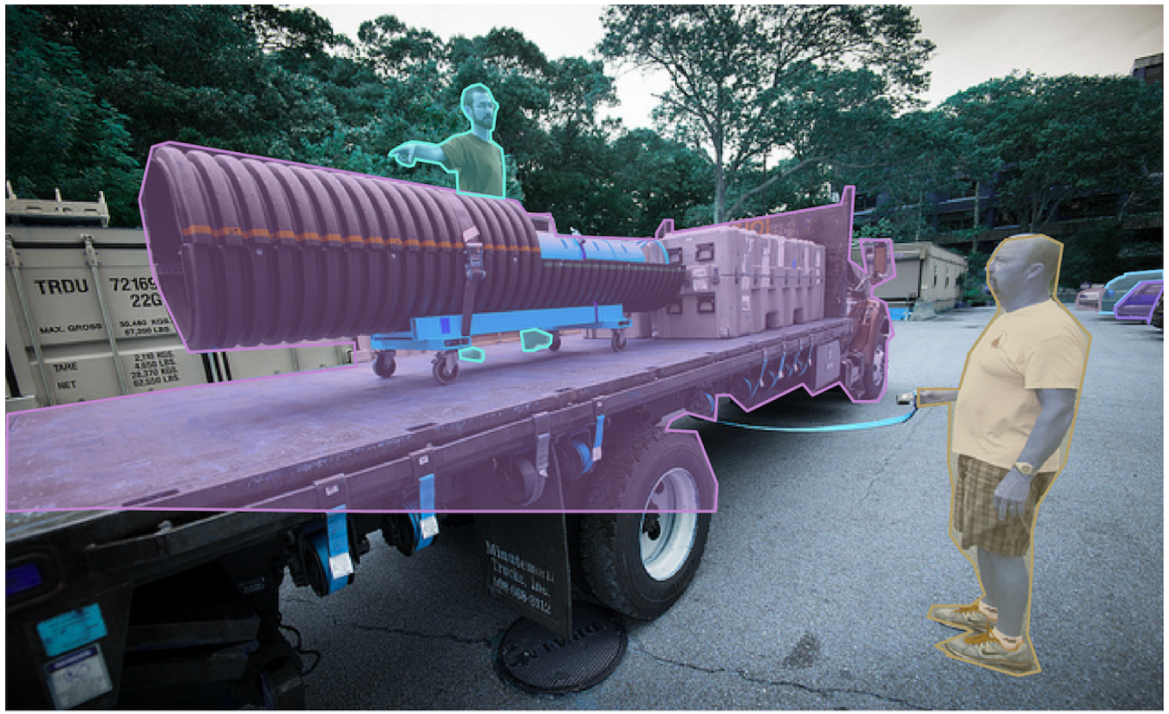


ground truth

segmentation results

Experience

3. The annotation quality may limit the performance.



ground truth

segmentation results

Experience

4. Engineering tricks matter.

Reproducing detection pipelines is not very easy.

- Some component works well in one DL framework, but it takes us long time to reimplement and debug it with another framework.
- It takes only 2 hours to implement an algorithm, but it may take 1 week to reproduce the performance reported in the paper.
- …

Experience

4. Engineering tricks matter.

There are traps everywhere.

- A wrong implementation of flip testing even decreases the mAP, the cause proves to be the rounding operation of bbox coordinates.
- A single pixel shift can lead to 1 point drop.

Experience

4. Engineering tricks matter.

reproduce existing methods
(20 days)

performance tuning
(30 days)

explore new ideas
(30 days)

We could do better if we already have a good codebase.

One more thing

Codebase

- **Comprehensive**
  - ☑ RPN
  - ☑ Mask R-CNN
  - ☑ Cascade R-CNN
  - ☐ More ··· ···
  - ☑ Fast/Faster R-CNN
  - ☑ FPN
  - ☑ RetinaNet
- **High performance**
  - ☑ Better performance
  - ☑ Optimized memory consumption
  - ☑ Faster speed
- **Handy to develop**
  - ☑ Written with PyTorch
  - ☑ Modular design

GitHub: mmdet

Thank you!