

# Mask Scoring R-CNN

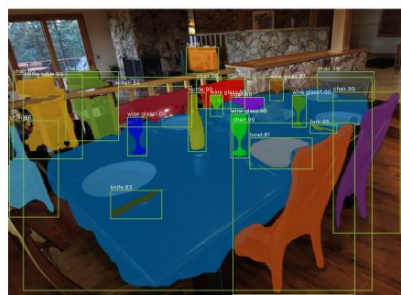
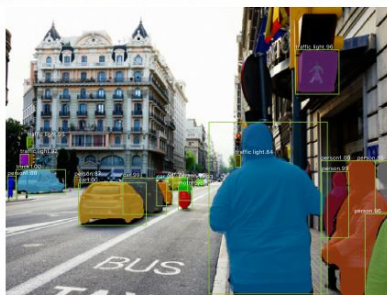
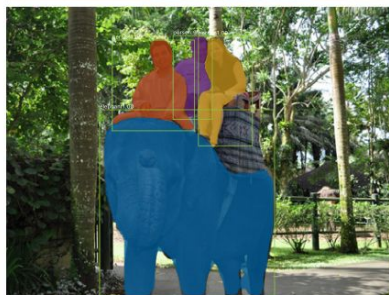
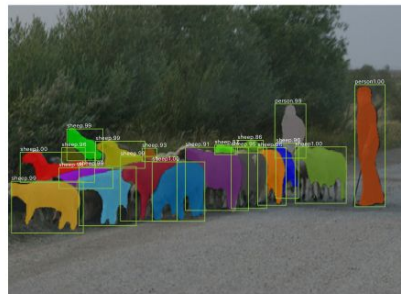
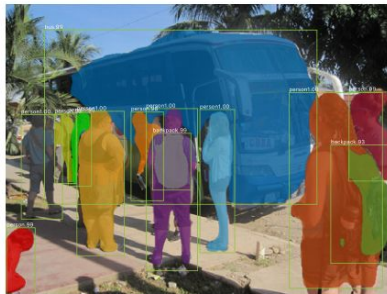
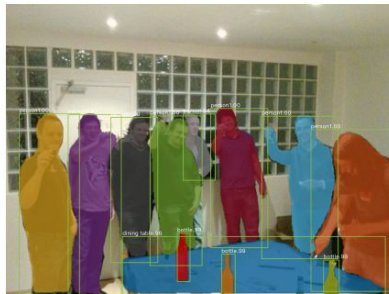
Zhaojin Huang <sup>†</sup> Lichao Huang <sup>‡</sup> Yongchao Gong <sup>‡</sup> Chang Huang <sup>‡</sup> Xinggang Wang <sup>†</sup>

<sup>†</sup> Institute of AI, School of EIC, Huazhong University of Science and Technology

<sup>‡</sup> Horizon Robotics Inc

# Instance Segmentation

Instance segmentation requires the correct **detection of all objects** in an image while also precisely **segmenting each instance**.



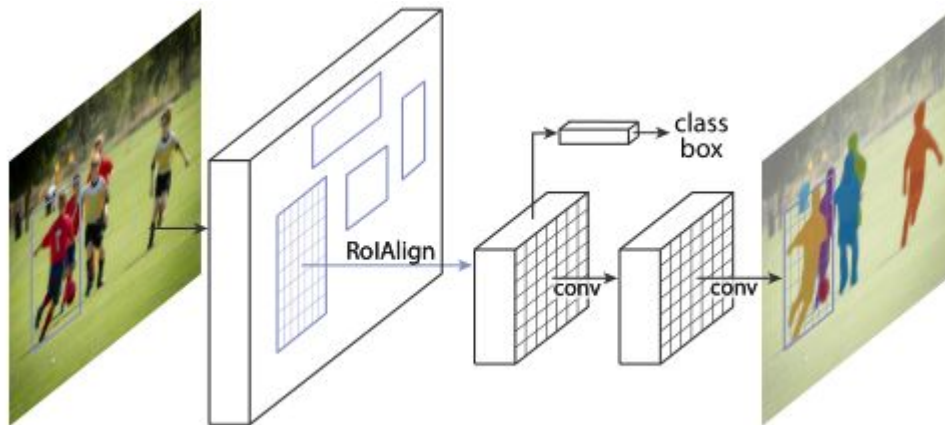
# Related Work

Instance segmentation:

1. Detection based methods:
  - FCIS [1] takes position-sensitive maps with inside/outside scores to generate the instance segmentation results.
  - Mask R-CNN [2] builds on the top of Faster R-CNN by adding an instance level semantic segmentation branch.
2. Segmentation based methods:
  - Liang et al. [3] uses spectral clustering to cluster the pixels.
  - Some works [4, 5] add boundary detection information during the clustering procedure.
  - Other works [6, 7, 8, 9] cluster instance by the learned embedding.

# Mask R-CNN

Mask R-CNN extends Faster R-CNN by adding a branch for predicting an object mask. It use classification score as instance segmentation score (called mask score).



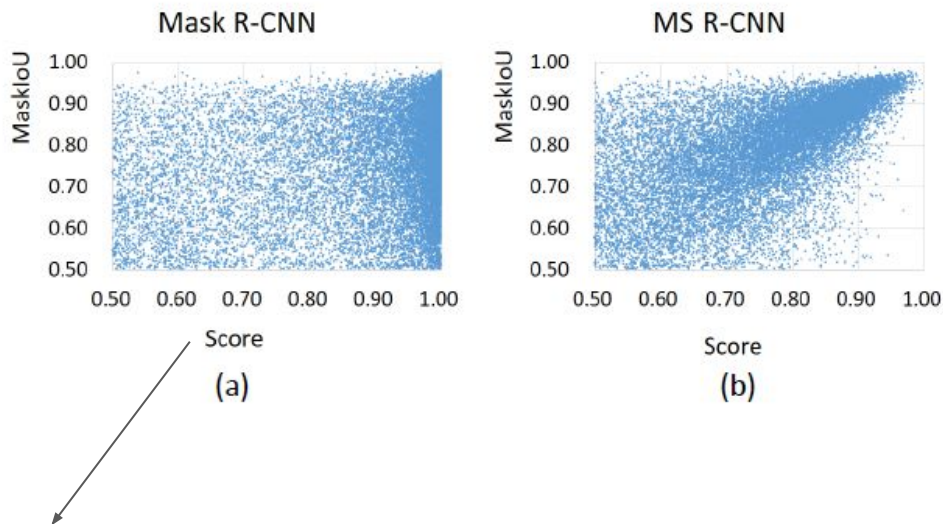
# Problem

The mask quality, IoU between the predicted mask and its ground truth mask (**called MaskIoU**), is usually **not well correlated** with the mask score.



# Motivation

Learning a calibrated mask score according to MaskIoU for every detection hypothesis.



The MaskIoU and mask score is not well correlated in Mask R-CNN.

# Related Work

IoU in detection:

- Fitness NMS [10] formulates box IoU prediction as a classification task.
- IoU-Net [11] regresses box IoU directly, and the predicted IoU was used for both NMS and bounding box refinement.

Our method is similar to IoU-Net. Here we list some differences with IoU-Net:

1. We use IoU for mask instead of box.
2. The IoU is used for correcting score instead of for box refinement or NMS.

# Method

Mask scoring:

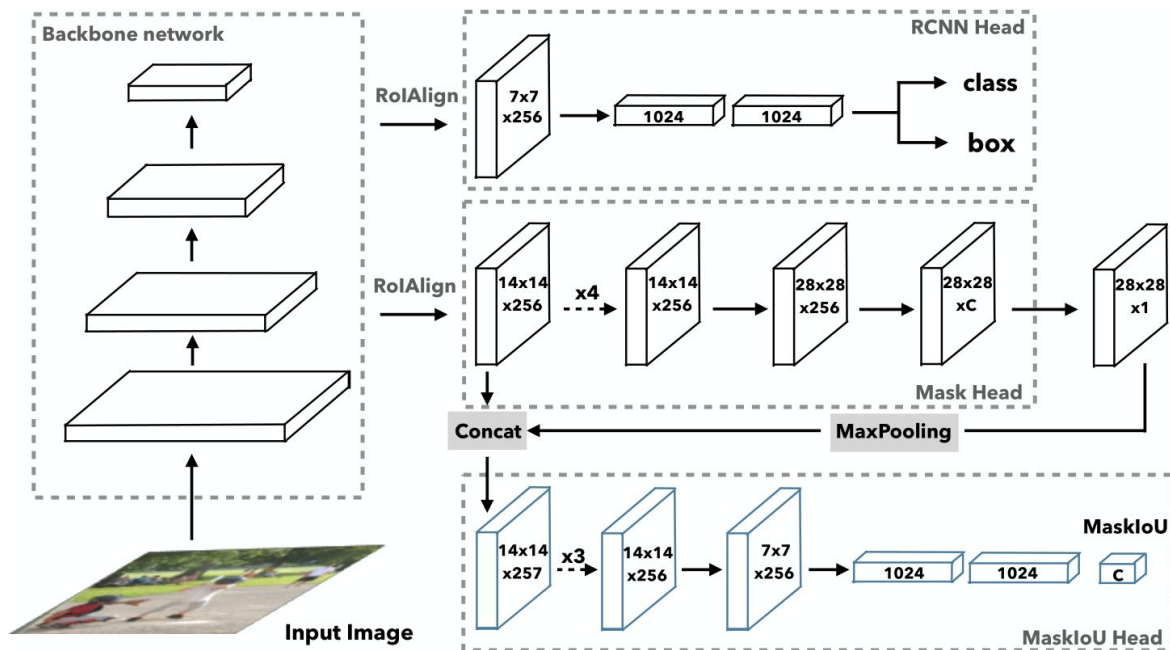
Decompose the mask score learning task into:

- **mask classification:** The mask classification score can directly take the corresponding classification score from R-CNN stage.
- **IoU regression.** The IoU can be learned by our propose MaskIoU head.



# Method

Network: extends Mask R-CNN by adding a branch called MaskIoU head for predicting MaskIoU.



# Method

Training:

- **Training samples for MaskIoU head:** the same with the training samples of the Mask head in Mask R-CNN.
- **Training targets:** the IoU between the predicted mask and its matched ground truth.
- **Training loss:** L2 loss for regression and the loss weight is set to 1.

# Method

Inference:

1. Selecting top-k (i.e.  $k = 100$ ) scoring boxes after SoftNMS.
2. The top-k boxes are fed into the Mask head to generate multi-class masks.
3. Feed the top-k target masks to MaskIoU head for predicting MaskIoU.
4. The predicted MaskIoU is multiplied with classification score, to get the new calibrated mask score as the final mask confidence.

1, 2 are the standard Mask R-CNN inference procedure.

# Quantitative Results

Results in different backbone networks (ResNet/18/50/101).

Backbone	MaskIoU head	AP <sub>m</sub>	AP <sub>m</sub> @0.5	AP <sub>m</sub> @0.75	AP <sub>b</sub>	AP <sub>b</sub> @0.5	AP <sub>b</sub> @0.75
ResNet-18 FPN	✓	27.7	46.9	29.0	31.2	50.4	33.2
		29.3	46.9	31.3	31.5	50.8	33.5
ResNet-50 FPN	✓	34.5	55.8	36.7	38.6	59.2	42.5
		36.0	55.8	38.8	38.6	59.2	42.5
ResNet-101 FPN	✓	36.6	58.6	39.0	41.3	61.7	45.9
		38.2	58.4	41.5	41.4	61.8	46.3

We can get improvement in different backbone network!

# Quantitative Results

Results in different frameworks (Faster R-CNN/FPN/DCN+FPN).

Backbone	MaskIoU head	FPN	DCN	AP <sub>m</sub>	AP <sub>m</sub> @0.5	AP <sub>m</sub> @0.75	AP <sub>b</sub>	AP <sub>b</sub> @0.5	AP <sub>b</sub> @0.75
ResNet-101	✓	✓		33.9	53.9	36.2	38.6	57.3	42.8
				35.0	54.0	37.7	38.7	57.4	43.0
				36.6	58.6	39.0	41.3	61.7	45.9
	✓	✓	✓	38.2	58.4	41.5	41.4	61.8	46.3
		✓		37.7	60.3	40.0	42.9	63.4	47.8
	✓	✓		39.1	60.0	42.4	43.1	63.5	47.7

We can get improvement in different framework!

# Quantitative Results

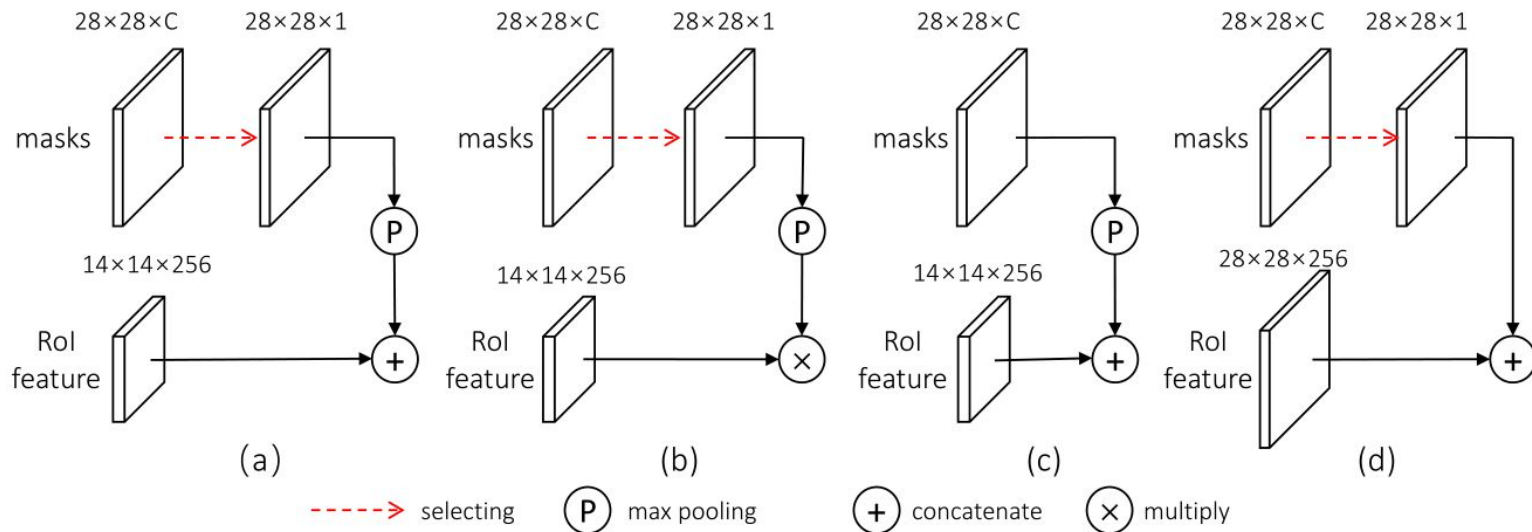
Comparing different instance segmentation methods on COCO 2017 test-dev.

Method	Backbone	AP	AP@0.5	AP@0.75	AP <sub>S</sub>	AP <sub>M</sub>	AP <sub>L</sub>
MNC [7]	ResNet-101	24.6	44.3	24.8	4.7	25.9	43.6
FCIS [23]	ResNet-101	29.2	49.5	-	-	-	-
FCIS+++ [23]	ResNet-101	33.6	54.5	-	-	-	-
Mask R-CNN [15]	ResNet-101	33.1	54.9	34.8	12.1	35.6	51.1
Mask R-CNN [15]	ResNet-101 FPN	35.7	58.0	37.8	15.5	38.1	52.4
Mask R-CNN [15]	ResNeXt-101 FPN	37.1	60.0	39.4	16.9	39.9	53.5
MaskLab [3]	ResNet-101	35.4	57.4	37.4	16.9	38.3	49.2
MaskLab+ [3]	ResNet-101	37.3	59.8	36.6	19.1	40.5	50.6
MaskLab+ [3]	ResNet-101 (JET)	38.1	61.1	40.4	19.6	41.6	51.4
Mask R-CNN	ResNet-101	34.3	55.0	36.6	13.2	36.4	52.2
MS R-CNN		35.4	54.9	38.1	13.7	37.6	53.3
Mask R-CNN	ResNet-101 FPN	37.0	59.2	39.5	17.1	39.3	52.9
MS R-CNN		38.3	58.8	41.5	17.8	40.4	54.4
Mask R-CNN	ResNet-101 DCN+FPN	38.4	61.2	41.2	18.0	40.5	55.2
MS R-CNN		39.6	60.7	43.1	18.8	41.5	56.2

# Ablation Study

The design choices of MaskIoU head input:

Setting	AP	AP@0.5	AP@0.75
Mask R-CNN baseline	27.7	46.9	29.0
(a) Target mask + RoI	29.3	46.9	31.3
(b) Target mask $\times$ RoI	29.1	46.6	30.9
(c) All masks + RoI	29.1	46.6	30.8
(d) Target mask + HR RoI	29.1	46.7	31.1



# Ablation Study

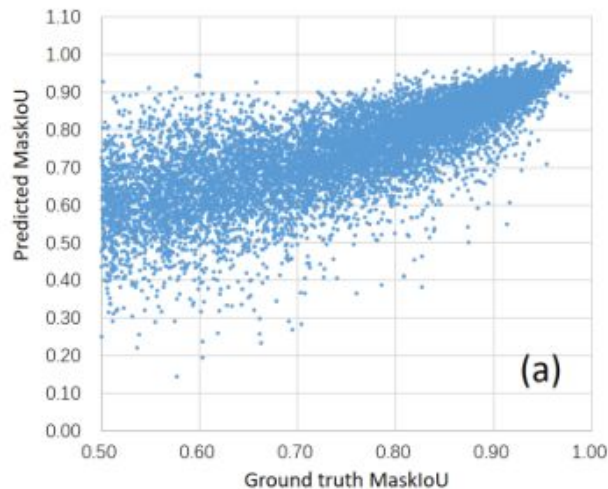
How to select training samples: we use the samples whose MaskIoU are larger than  $\tau$  to train the MaskIoU head.

Threshold	AP	AP@0.5	AP@0.75
$\tau = 0.0$	29.3	46.9	31.3
$\tau = 0.3$	29.2	46.6	31.1
$\tau = 0.5$	29.0	46.5	30.9
$\tau = 0.7$	28.8	46.9	30.5

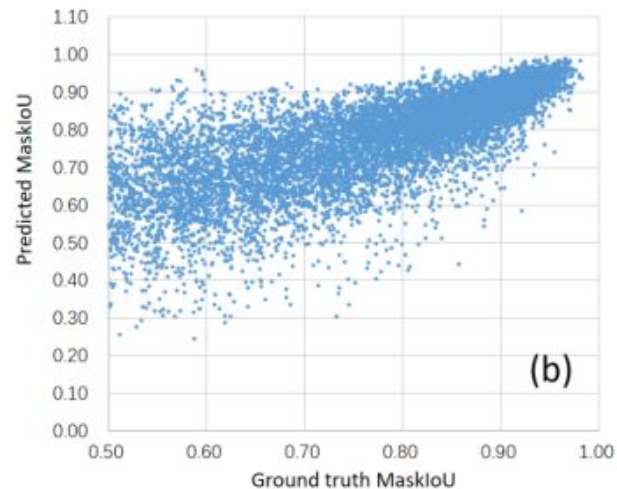


# Discussion

The quality of the predicted MaskIoU: the correlation coefficient are both about 0.74.



ResNet-18 FPN



ResNet-101 DCN+FPN

# Discussion

The upper bound performance of MS RCNN: use the ground truth MaskIoU to replace the predicted MaskIoU when the ground truth MaskIoU larger than 0.

Method	Backbone	AP
Mask R-CNN	ResNet-18 FPN	27.7
MS R-CNN		29.3
MS R-CNN*		31.5
Mask R-CNN	ResNet-101 DCN+FPN	37.7
MS R-CNN		39.1
MS R-CNN*		41.7

# Discussion

- **FLOPs**: our MaskIoU head has about 0.39G FLOPs while Mask head has about 0.53G FLOPs for each proposal.
- **Running time**: the testing speed (sec./image) of MS R-CNN and Mask R-CNN is almost the same. ResNet-18 FPN: both about 0.132. ResNet-101 DCN+FPN: both about 0.202.

# Codes

Github: [https://github.com/zjhuang22/maskscoring\\_rcnn](https://github.com/zjhuang22/maskscoring_rcnn) (nearly 1000 stars)

# Thanks

# References

- [1] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully convolutional instance-aware semantic segmentation. In CVPR, 2017
- [2] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In ICCV, 2017
- [3] X. Liang, Y. Wei, X. Shen, J. Yang, L. Lin, and S. Yan. Proposal-free network for instance-level object segmentation. arXiv preprint arXiv:1509.02636.
- [4] L. Jin, Z. Chen, and Z. Tu. Object detection free instance segmentation with labeling transformations. arXiv preprint arXiv:1611.08991, 2016.
- [5] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. Instancecut: from edges to instances with multicut. In CVPR, 2017
- [6] B. De Brabandere, D. Neven, and L. Van Gool. Semantic instance segmentation with a discriminative loss function. arXiv preprint arXiv:1708.02551
- [7] A. Fathi, Z. Wojna, V. Rathod, P. Wang, H. O. Song, S. Guadarrama, and K. P. Murphy. Semantic instance segmentation via deep metric learning. arXiv preprint arXiv:1703.10277
- [8] A. W. Harley, K. G. Derpanis, and I. Kokkinos. Segmentation-aware convolutional networks using local attention masks. In CVPR, 2017
- [9] A. Newell, Z. Huang, and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. In NIPS, 2017
- [10] L. Tychsen-Smith and L. Petersson. Improving object localization with fitness nms and bounded iou loss. In CVPR, 2018.
- [11] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang. Acquisition of localization confidence for accurate object detection. In ECCV, 2018