MASK R-CNN

Presented by Cody Kidwell

Facebook AI Research (FAIR)

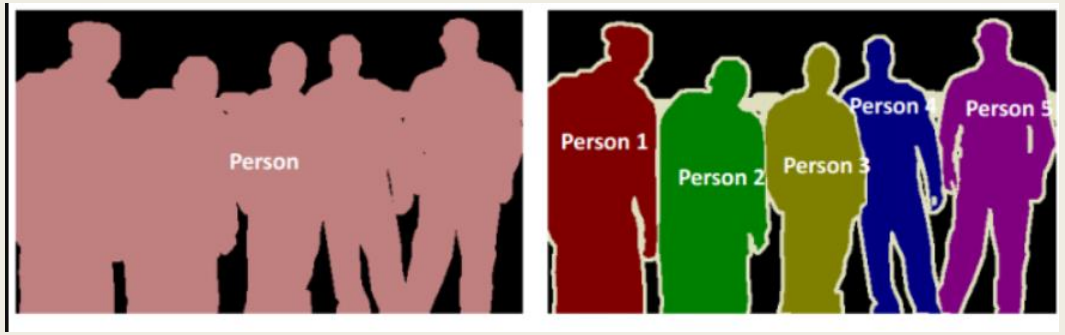
- Kaiming He
- Georgia Gkioxari
- Piotr Dollar
- Ross Girshick

- K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, 2017, pp. 2980-2988.

Introduction

- Their approach was designed to efficiently detect objects in an image while also, simultaneously generating a high-quality segmentation mask for each instance.
- At the time advances in object detection and semantic segmentation were being driven by powerful baseline architectures such as Fast R-CNN, Faster R-CNN and Full Convolutional Networks. Although, instance segmentation was not yet solved using these baseline architectures.
- These decided to take the same approach for their instance segmentation problem by extending the Faster R-CNN architecture. They did this by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition.
- They were able to do all this and get positive results, by only adding a small overhead, running at 5fps.

Extends Faster R-CNN

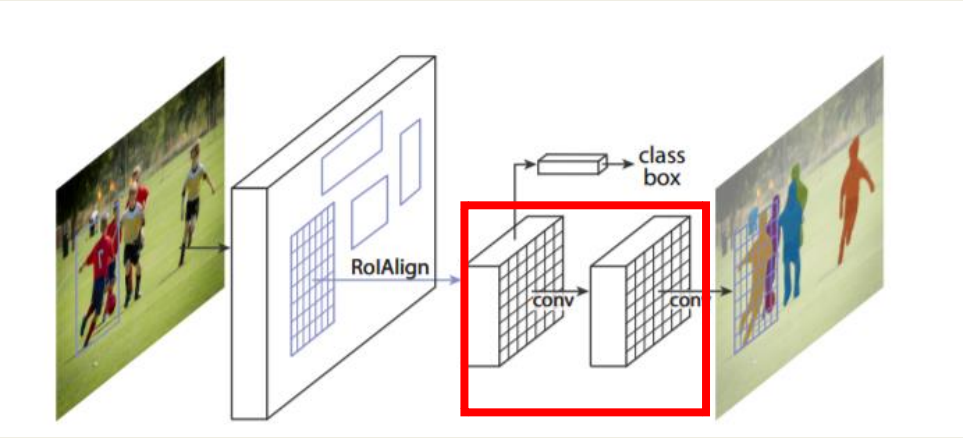


Semantic Segmentation Instance Segmentation

- When developing Mask R-CNN, Faster R-CNN was the state of art object detection architecture.
- Instead of using other more complex methods to achieve image segmentation, they show a method that builds upon Faster R-CNN.
- In parallel to the class label and bounding box offset, they create a new branch to the architecture that outputs the object mask. This branch is the mask branch.
- This new branch is a Fully Convolutional Network. This is to keep the spatial orientation of the pixels, unlike Fully Connected Layers.

Procedure

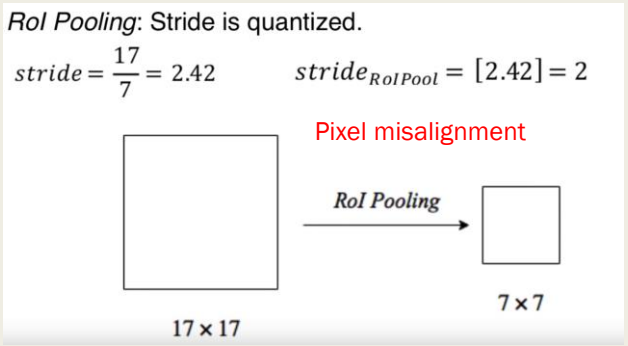
- Mask R-CNN takes the same two-stage procedure that Faster R-CNN takes.
- The first stage which is identical, is RPN (Region Proposal Network).
- The second stage, in parallel to predicting the class and box offset, also outputs the binary mask for each region of interest.
- This is different from most other systems at the time, where classification depends on the mask predictions.

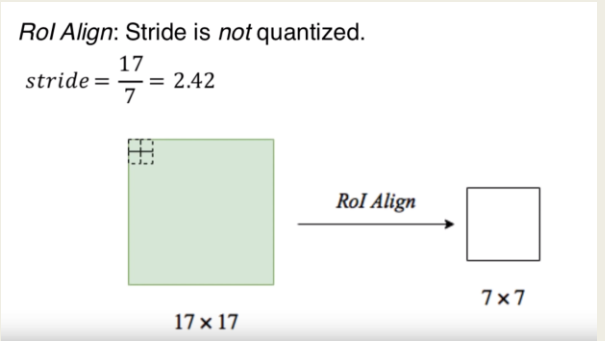


Loss

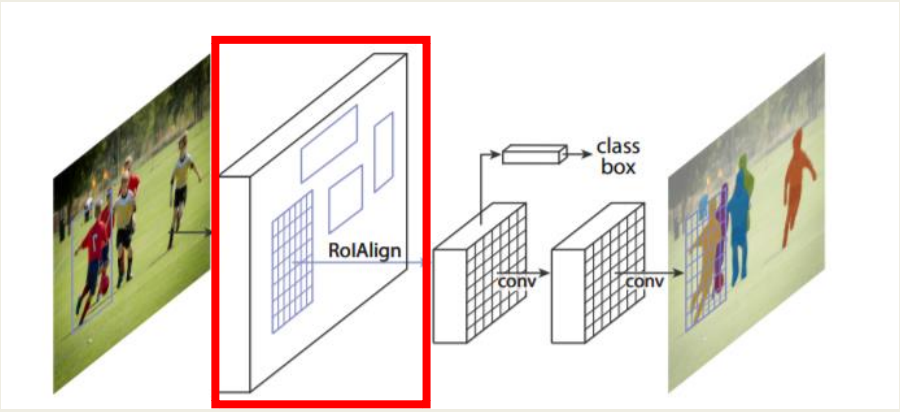
- To compute the mask during training, they define a multi-task loss on each region of interest. This is L = Lcls + Lbox + Lmask. Lcls and Lbox coming from Faster R-CNN.
- This new branch has a Km^2 dimensional output for each region of interest, which encodes the binary masks of resolution m*m, one for each of the K classes.
- Once they have this, they apply a per-pixel sigmoid and define Lmask as the average binary cross-entropy loss.
- There definition of Lmask allows for the network to generate masks for every class without competition among the classes. This separates mask and class prediction. This is different from most other cases where the class and mask prediction compete.
- Mask R-CNN has per-pixel sigmoid and binary loss, where others use, per-pixel SoftMax and multinomial cross-entropy loss.

RoIAlign





- Faster R-CNN used RoIPool, which was not designed for pixel-to-pixel alignment between inputs and outputs.
- To fix this alignment issue, they introduced a layer called RoIAlign that would preserve exact spatial locations.



- Minor change that made large impact. It improves mask accuracy my relative 10% to 50%.

- RoIAlign uses bilinear interpolation to computes the exact values of the input features at four regularly sampled locations within each region of interest, bin and aggregate the result (using max or average).
- They note that results are not sensitive to exact sampling locations, or even the number of points sampled, considering there was not quantization performed.
- They also tried RoIWarp but it did not address alignment issues. RoIAlign brought signification improvements as show in the data later presented.

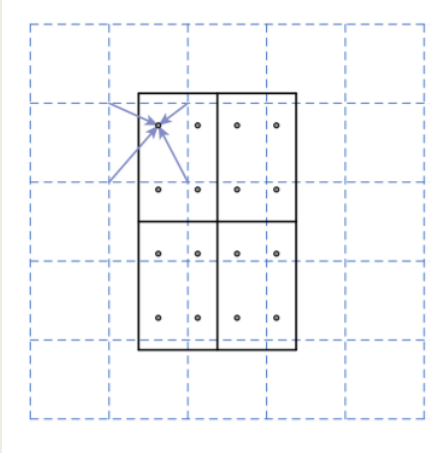
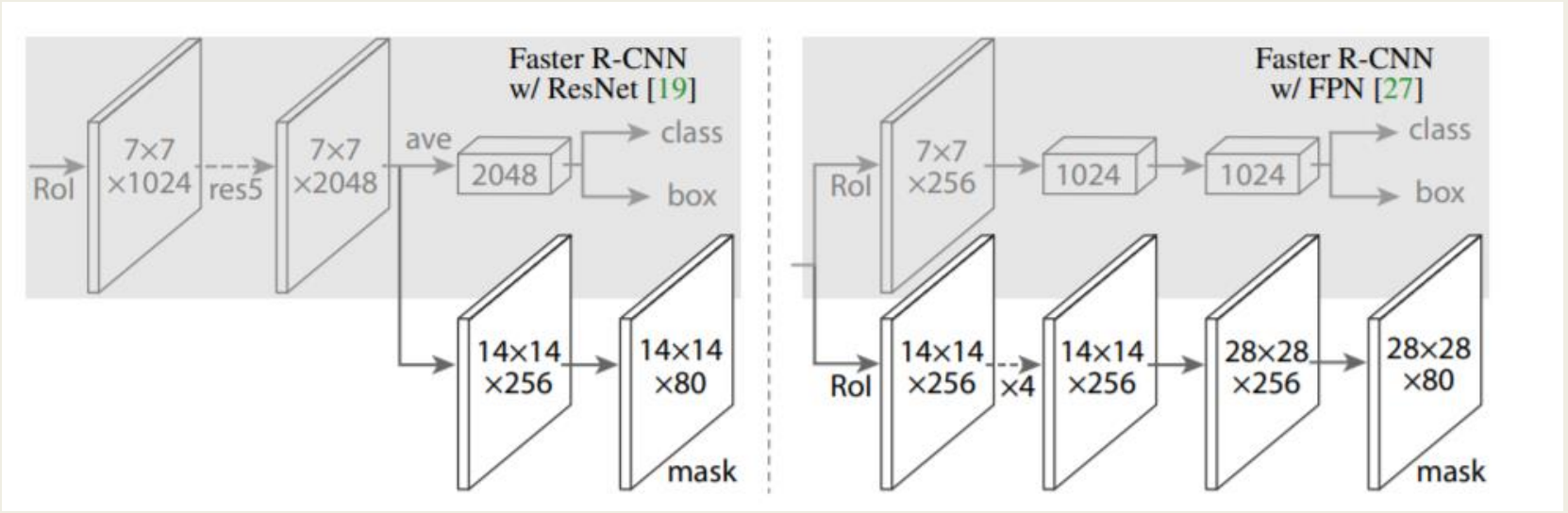


Figure 3. **RoIAlign:** The dashed grid represents a feature map, the solid lines an RoI (with 2×2 bins in this example), and the dots the 4 sampling points in each bin. RoIAlign computes the value of each sampling point by bilinear interpolation from the nearby grid points on the feature map. No quantization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

Architecture Approaches

- To address generality of their approach they test Mask R-CNN with multiple architectures as backbones.
- They used ResNeXt and ResNet with depths of 50 and 101 layers as a backbone, in addition to, Feature Pyramid Networks (FPN).
- With these two sets of backbones they created two similar network heads that were very similar to the fully convolutional mask prediction branch. Although with slight differences.

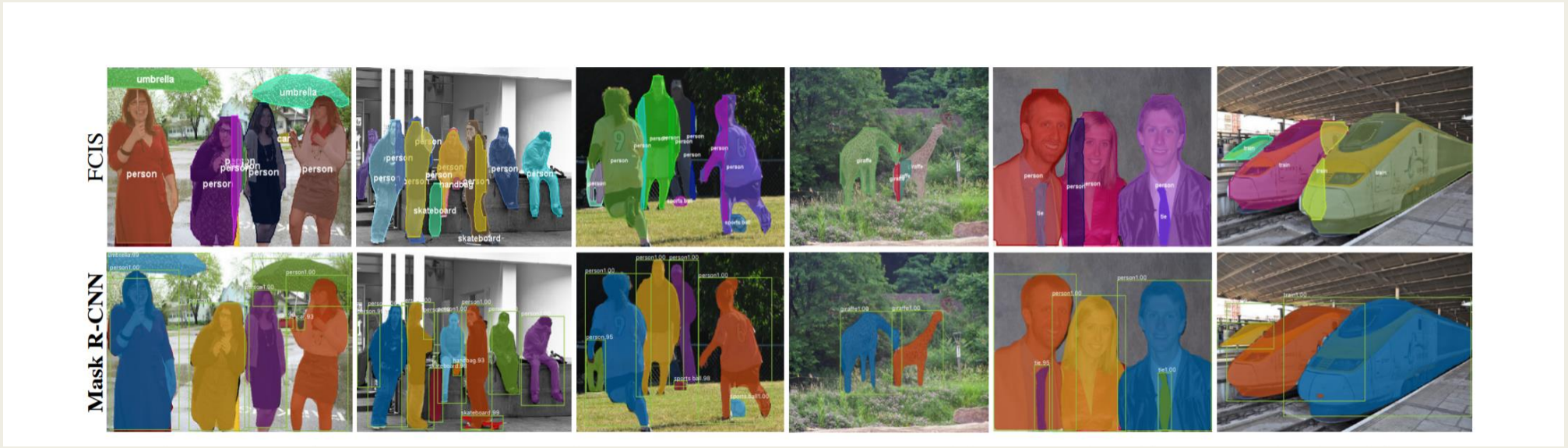


Training

- As in Fast R-CNN, a region of interest is considered positive if it has intersection over union with a ground-truth box has at least 0.5, otherwise it is negative. The mask loss Lmask is defined only on positive region of interests. The mask target is the intersection between a region of interest and its associated ground-truth mask.
- They resized their images' shorter side to 800 pixels.
- Trained on a GPU with mini batches of 2 images per GPU. They trained on a total of 8 GPUs(16 batch size) for 160,000 iterations, with a learning rate of 0.02 which decreases by 10 at the 120,000th iteration. They also used a weight decay of 0.0001 and momentum of 0.9.

Experiments

- Main dataset was MS COCO, which was 80 classes and 80k train images.



Fully Convolutional Instance-aware Semantic Segmentation (FCIS) exhibits systematic artifacts on overlapping objects.

Architecture Backbone Results

	backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
MNC [10]	ResNet-101-C4	24.6	44.3	24.8	4.7	25.9	43.6
FCIS [26] +OHEM	ResNet-101-C5-dilated	29.2	49.5	-	7.1	31.3	50.0
FCIS+++ [26] +OHEM	ResNet-101-C5-dilated	33.6	54.5	-	-	-	-
Mask R-CNN	ResNet-101-C4	33.1	54.9	34.8	12.1	35.6	51.1
Mask R-CNN	ResNet-101-FPN	35.7	58.0	37.8	15.5	38.1	52.4
Mask R-CNN	ResNeXt-101-FPN	**37.1**	**60.0**	**39.4**	**16.9**	**39.9**	**53.5**

More Results

net-depth-features	AP	AP_{50}	AP_{75}
ResNet-50-C4	30.3	51.2	31.5
ResNet-101-C4	32.7	54.2	34.3
ResNet-50-FPN	33.6	55.2	35.3
ResNet-101-FPN	35.4	57.3	37.5
ResNeXt-101-FPN	**36.7**	**59.5**	**38.9**

(a) **Backbone Architecture**: Better backbones bring expected gains: deeper networks do better, FPN outperforms C4 features, and ResNeXt improves on ResNet.

	AP	AP_{50}	AP_{75}
softmax	24.8	44.1	25.1
sigmoid	**30.3**	**51.2**	**31.5**
	+5.5	*+7.1*	*+6.4*

(b) **Multinomial *vs.* Independent Masks** (ResNet-50-C4): *Decoupling* via per-class binary masks (sigmoid) gives large gains over multinomial masks (softmax).

	align?	bilinear?	agg.	AP	AP_{50}	AP_{75}
RoIPool [12]			max	26.9	48.8	26.4
RoIWarp [10]		✓	max	27.2	49.2	27.1
		✓	ave	27.1	48.9	27.1
RoIAlign	✓	✓	max	**30.2**	**51.0**	**31.8**
	✓	✓	ave	**30.3**	**51.2**	**31.5**

(c) **RoIAlign** (ResNet-50-C4): Mask results with various RoI layers. Our RoIAlign layer improves AP by ~3 points and AP_{75} by ~5 points. Using proper alignment is the only factor that contributes to the large gap between RoI layers.

	AP	AP_{50}	AP_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
RoIPool	23.6	46.5	21.6	28.2	52.7	26.9
RoIAlign	**30.9**	**51.8**	**32.1**	**34.0**	**55.3**	**36.4**
	+7.3	*+ 5.3*	*+10.5*	*+5.8*	*+2.6*	*+9.5*

(d) **RoIAlign** (ResNet-50-**C5**, *stride 32*): Mask-level and box-level AP using *large-stride* features. Misalignments are more severe than with stride-16 features (Table 2c), resulting in big accuracy gaps.

	mask branch	AP	AP_{50}	AP_{75}
MLP	fc: 1024→1024→80·28^2	31.5	53.7	32.8
MLP	fc: 1024→1024→1024→80·28^2	31.5	54.0	32.6
FCN	conv: 256→256→256→256→256→80	**33.6**	**55.2**	**35.3**

(e) **Mask Branch** (ResNet-50-FPN): Fully convolutional networks (FCN) *vs.* multi-layer perceptrons (MLP, fully-connected) for mask prediction. FCNs improve results as they take advantage of explicitly encoding spatial layout.

Extra: Human Pose Estimation

- Their framework can be easily extended to human pose estimation.
- They note that minimal domain knowledge for human pose is exploited by their system. They also note that it was to demonstrate the generality of Mask R-CNN.
- They make minor modifications to the segmentation system. Which includes minimizing the cross-entropy loss over an m^2-way softmax output.
- Results from this show 0.9 points higher from 2016’s keypoint detection winner.

Thank You