The surprising impact of mask-head architecture on novel class segmentation

Vighnesh Birodkar, Zhichao Lu, Siyang Li, Vivek Rathod, Jonathan Huang

Google

ICCV 2021

https://arxiv.org/abs/2104.00613

Presenter: Minho Park

Task

- Partially supervised instance segmentation
- Collecting groundtruth masks can takes > 10× more time than bounding box annotations.
  - In COCO, mask annotations required about 80 sec whereas bounding box annotations need 7 sec. (Dim P. Papadopoulos et al.)

Extreme clicking for efficient object annotation, Dim P. Papadopoulos et al., https://arxiv.org/abs/1708.02750

Partially supervised instance segmentation

- All classes have bounding box annotations but only a subset of classes have mask annotations. (E.g., PASCAL VOC → Non-VOC)

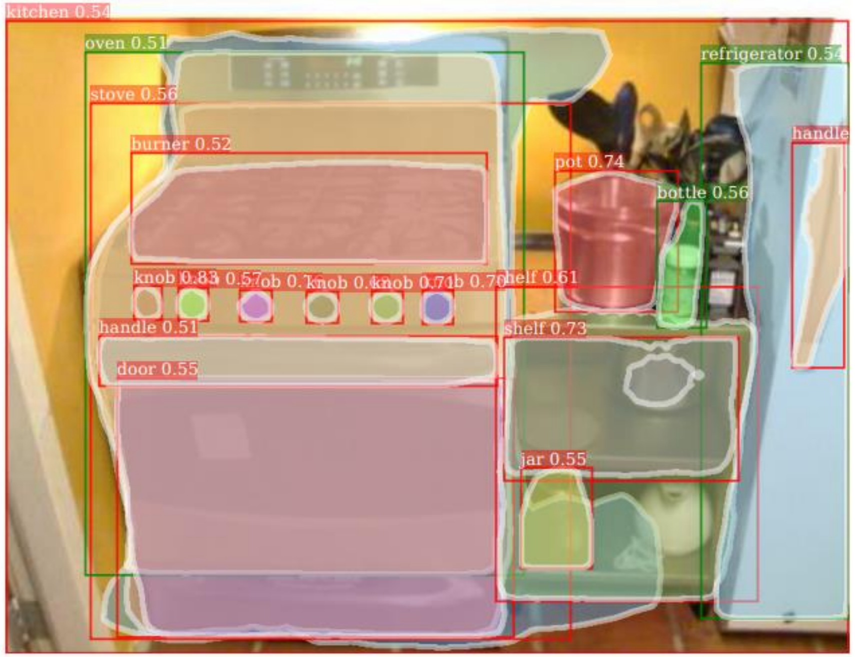
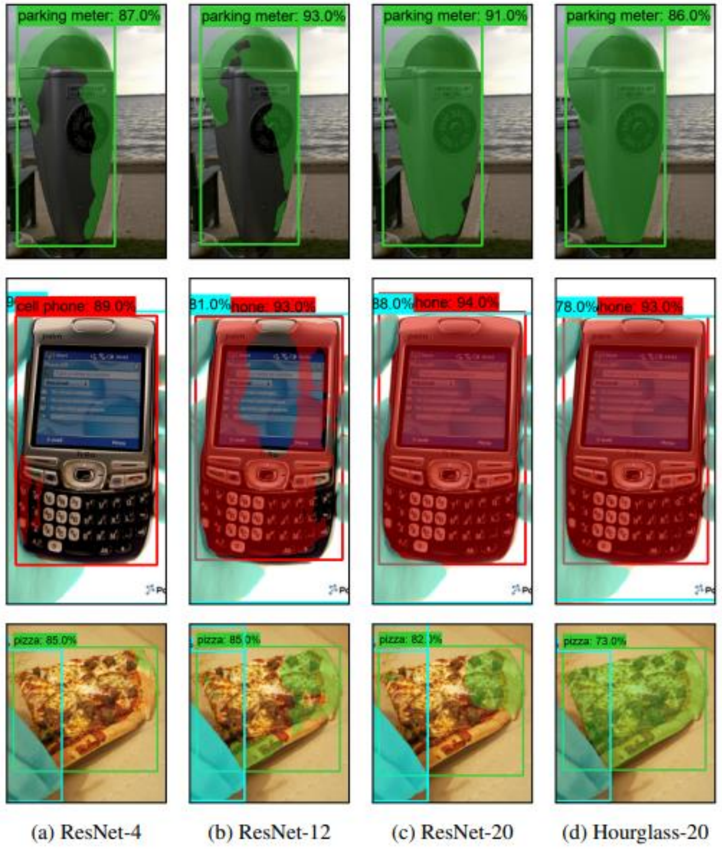


Figure 1. **We explore training instance segmentation models with partial supervision**: a subset of classes (green boxes) have instance mask annotations during training; the remaining classes (red boxes) have only bounding box annotations. This image shows output from our model trained for 3000 classes from Visual Genome, using mask annotations from only 80 classes in COCO.

Learning to Segment Every Thing, Ronghang Hu et al., https://arxiv.org/abs/1711.10370

Objective

- Strong mask generalization effect
    - Ability of mask-heads to generalize to unseen classes
    - E.g., in figure 1, the mask-head architecture never having seen masks from the 'parking meter', 'pizza' or 'mobile phone' class



(a) ResNet-4 (b) ResNet-12 (c) ResNet-20 (d) Hourglass-20

Figure 1: The effect of mask-head architecture on mask predictions for unseen classes.

Naïve Baseline

- Adapt Mask R-CNN to produce class-agnostic masks.
  - Mask R-CNN with a class-agnostic FCN
- Perform abysmally on unseen classes
  - E.g., on the standard partially supervised COCO benchmark, it achieves < 20% mask mAP on unseen classes vs > 40% on seen. (Ronghang Hu et al.)
  - Thus previous approaches have used. E.g., offline-trained shape priors. (Weicheng Kuo et al.)

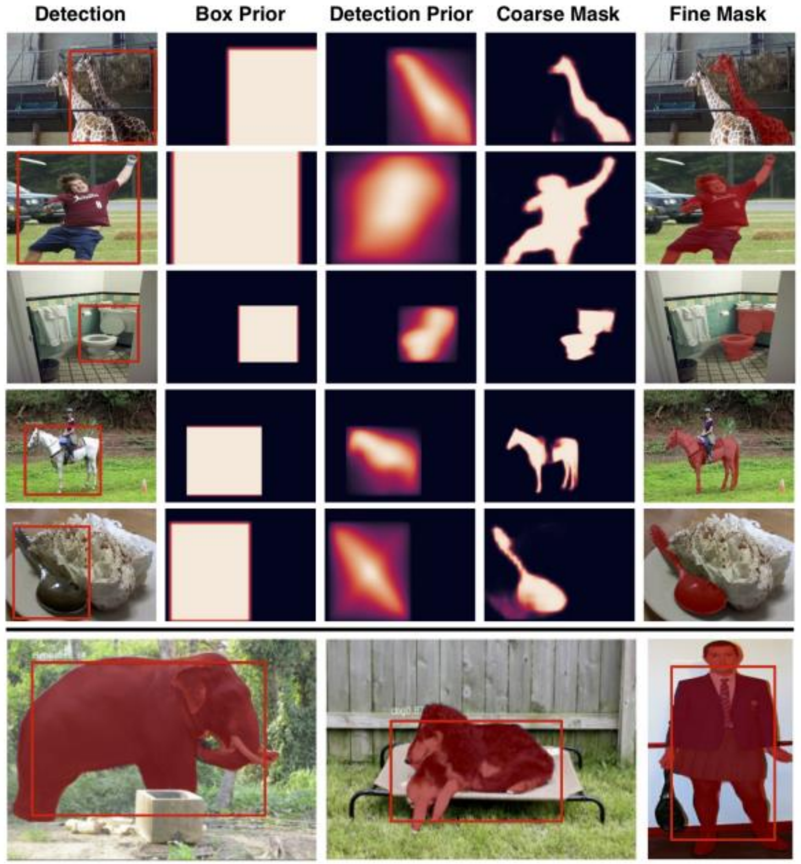


Figure 1: ShapeMask instance segmentation is designed to learn the shape of objects by refining object shape priors.

Learning to Segment Every Thing, Ronghang Hu et al., https://arxiv.org/abs/1711.10370
ShapeMask: Learning to Segment Novel Objects by Refining Shape Priors, Weicheng Kuo et al., https://arxiv.org/abs/1904.03239

RoIAlign

- Removes the harsh quantization of RoIPool
- Use bilinear interpolation to compute the exact values of the input features
- 4 sampling points in each bin
- Aggregate the result (using max or average)

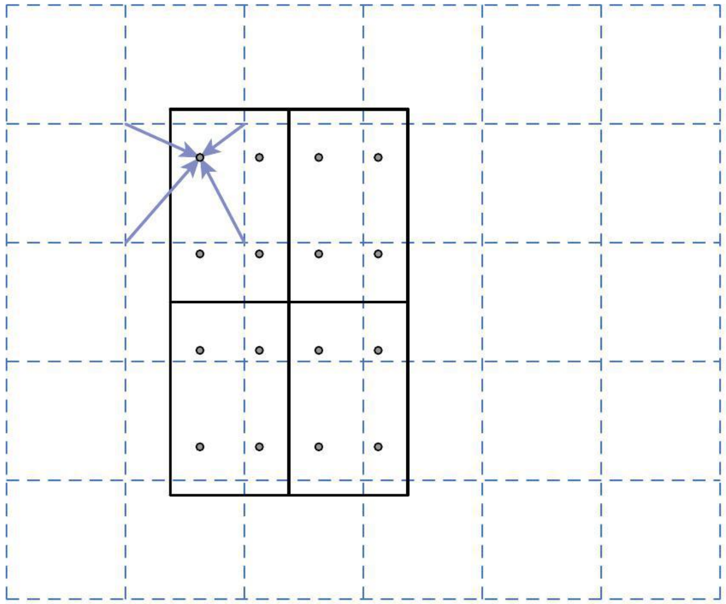


Figure 3. RoIAlign

Mask R-CNN, Kaiming He et al., https://arxiv.org/abs/1703.06870

Anchor-based Methods

- Mask R-CNN

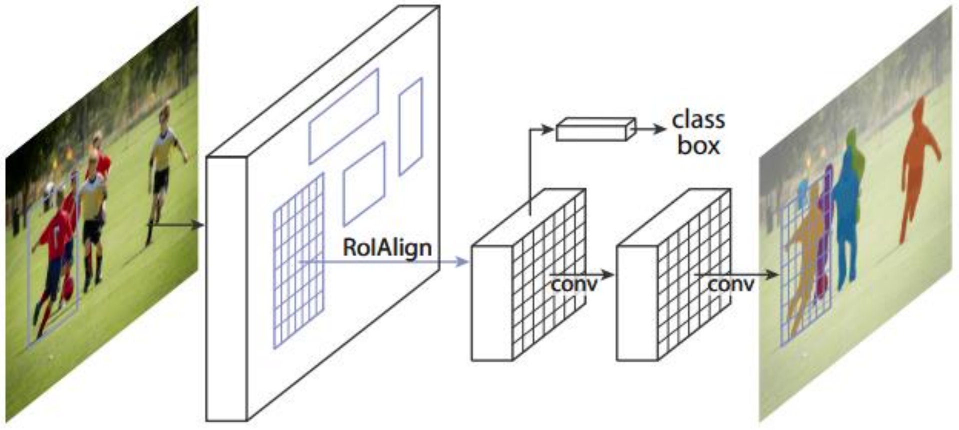


Figure 1. The **Mask R-CNN** framework for instance segmentation.

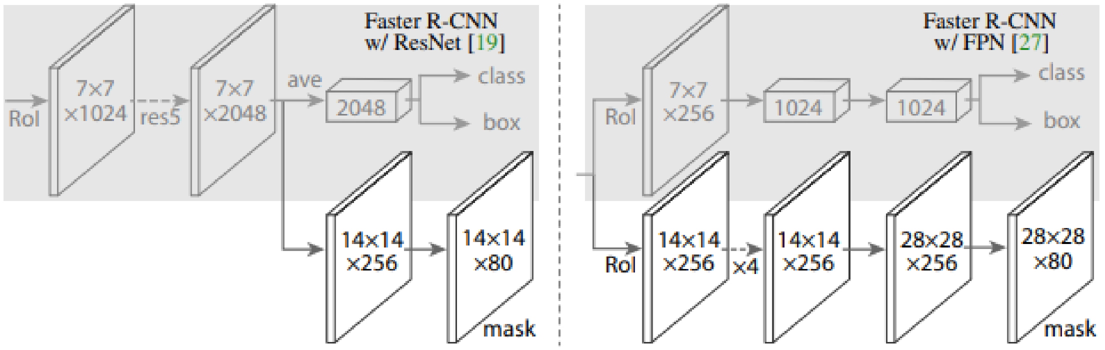


Figure 4. Head Architecture

Mask R-CNN, Kaiming He et al., https://arxiv.org/abs/1703.06870

Anchor-free Methods

- CenterNet

$$L_k = \frac{-1}{N} \sum_{xyc} \begin{cases} (1-\hat{Y}_{xyc})^{\alpha} \log(\hat{Y}_{xyc}) & \text{if } Y_{xyc} = 1 \\ \begin{array}{c} (1-Y_{xyc})^{\beta} (\hat{Y}_{xyc})^{\alpha} \\ \log(1-\hat{Y}_{xyc}) \end{array} & \text{otherwise} \end{cases} \qquad (1)$$

$$L_{off} = \frac{1}{N} \sum_{p} \left| \hat{O}_{\tilde{p}} - \left(\frac{p}{R} - \tilde{p} \right) \right| . \qquad (2)$$

$$L_{size} = \frac{1}{N} \sum_{k=1}^{N} \left| \hat{S}_{p_k} - s_k \right| . \qquad (3)$$

$$L_{det} = L_k + \lambda_{size} L_{size} + \lambda_{off} L_{off} . \qquad (4)$$

  - $Y \in [0,1]^{\frac{W}{R}\times\frac{H}{R}\times C}$: Keypoint (Center of object)
  - $O \in \mathcal{R}^{\frac{W}{R}\times\frac{H}{R}\times 2}$: Offset
  - $S \in \mathcal{R}^{\frac{W}{R}\times\frac{H}{R}\times 2}$: Size (Height, Width)

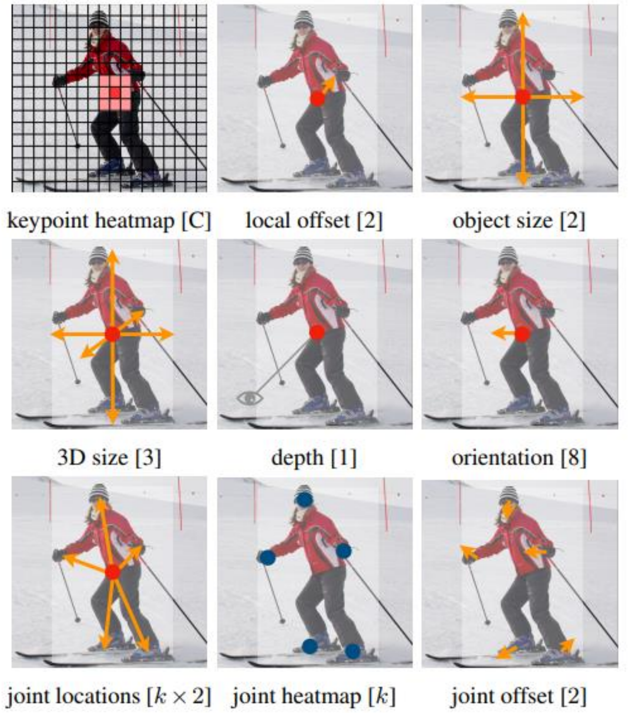


Figure 4: Outputs of our network for different tasks: *top* for object detection, *middle* for 3D object detection, *bottom:* for pose estimation. All modalities are produced from a common backbone, with a different 3 × 3 and 1 × 1 output convolutions separated by a ReLU. The number in brackets indicates the output channels. See section 4 for details.

Objects as Points, Xingyi Zhou et al., https://arxiv.org/abs/1904.07850

Anchor-free Methods

- FCOS: Classification + Center-neess + Regression
  - Not rely on predefined anchor boxes
  - $\text{Centerness} = \sqrt{\frac{\min(l,r)}{\max(l,r)} \times \frac{\min(t,b)}{\max(t,b)}}$

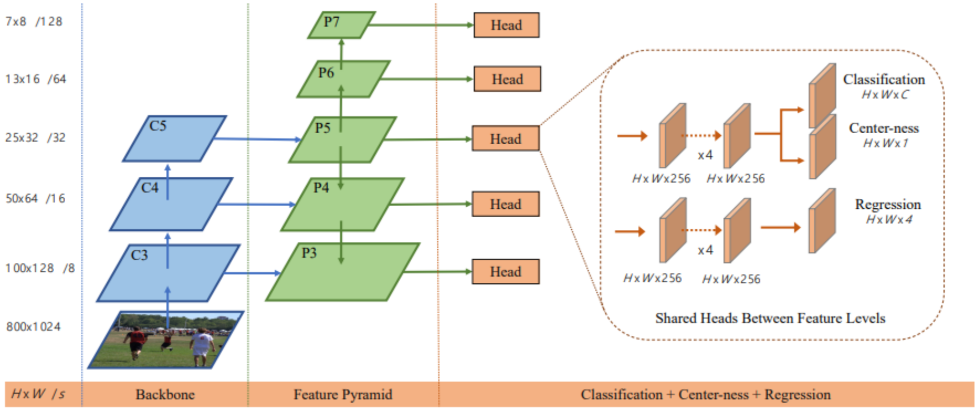


Figure 2 – The network architecture of FCOS, where C3, C4, and C5 denote the feature maps of the backbone network and P3 to P7 are the feature levels used for the final prediction. $H \times W$ is the height and width of feature maps. '/s' ($s = 8, 16, ..., 128$) is the down-sampling ratio of the feature maps at the level to the input image. As an example, all the numbers are computed with an 800×1024 input.

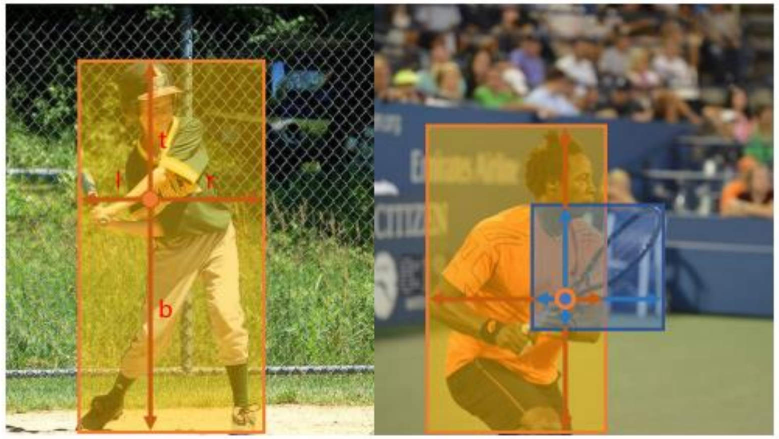


Figure 1 – As shown in the left image, FCOS works by predicting a 4D vector (l, t, r, b) encoding the location of a bounding box at each foreground pixel (supervised by ground-truth bounding box information during training). The right plot shows that when a location residing in multiple bounding boxes, it can be ambiguous in terms of which bounding box this location should regress.

FCOS: Fully Convolutional One-Stage Object Detection, Zhi Tian et al., https://arxiv.org/abs/1904.01355

Crop-then-segment Instance Segmentation

- Deep-MARC
  - Deep Mask-heads Above R-CNN
- Deep-MAC
  - Deep Mask-heads Above CenterNet
- Without a detector
  - Deep Mask-heads Above ground truth bounding boxes

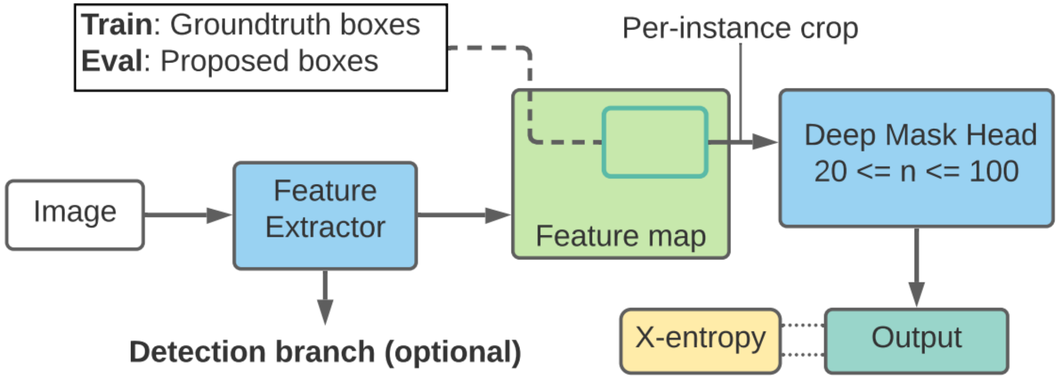


Figure 2: Diagram of the crop-then-segment instance segmentation model family.

Additional Tricks

- For improve stability of some mask-heads, concatenate below two embeddings.

1. Instance Embedding
   - Add an additional head to the backbone that predicts a per-pixel embedding
   - This helps condition the mask-head on a particular instance and disambiguate it from others.
2. Coordinate Embedding
   - Inspired by CoordConv, we add a 32 × 32 × 2 tensor holding normalized (x, y) coordinates relative to the bounding box b.

Two choices

- In order to achieve **strong mask generalization** …

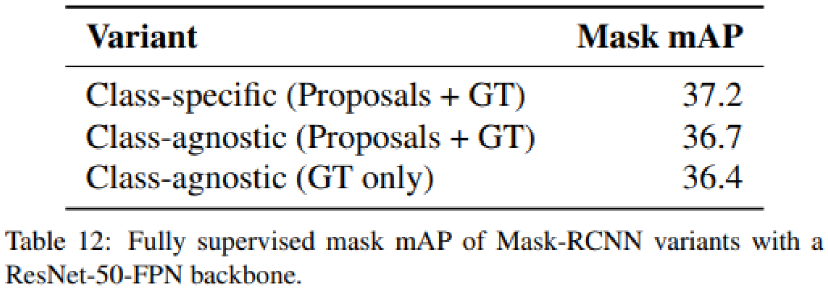
1. Whether to crop to ground truth boxes or both ground truth boxes and proposals when training the mask heads (of the detector-based models).
    - Train with only ground truth
2. Which mask-head architecture to use.
    - Use significantly deeper mask head architectures

Experimental Setup

- 80 COCO categories
  - = 20 Pascal VOC categories + 60 non-VOC categories
- Train with 20 Pascal VOC categories having instance masks
- Evaluate performance on the 60 non-VOC categories
- Called VOC-Masks-Only

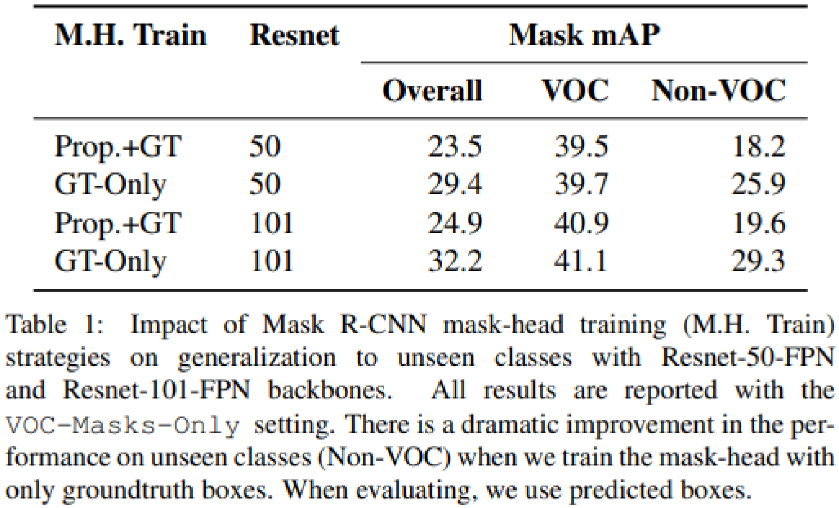
Cropping to Only Ground Truth Boxes

- In full supervision setups, train with Proposals + GT is slightly better than GT-Only.
- However, in partially supervised setups, training with GT-Only dramatically improves performance for the Non-VOC (unseen) classes.



Variant	Mask mAP
Class-specific (Proposals + GT)	37.2
Class-agnostic (Proposals + GT)	36.7
Class-agnostic (GT only)	36.4

Table 12: Fully supervised mask mAP of Mask-RCNN variants with a ResNet-50-FPN backbone.

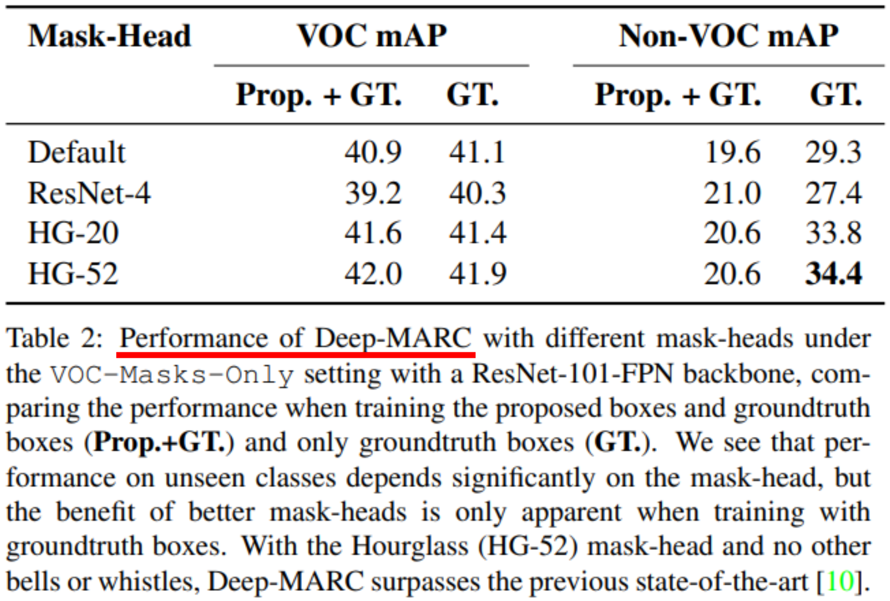


M.H. Train	Resnet	Mask mAP		
		Overall	VOC	Non-VOC
Prop.+GT	50	23.5	39.5	18.2
GT-Only	50	29.4	39.7	25.9
Prop.+GT	101	24.9	40.9	19.6
GT-Only	101	32.2	41.1	29.3

Table 1: Impact of Mask R-CNN mask-head training (M.H. Train) strategies on generalization to unseen classes with Resnet-50-FPN and Resnet-101-FPN backbones. All results are reported with the `VOC-Masks-Only` setting. There is a dramatic improvement in the performance on unseen classes (Non-VOC) when we train the mask-head with only groundtruth boxes. When evaluating, we use predicted boxes.

Going Deeper with Mask Heads

- Deep-MARC
- The difference between worst and best case in seen classes is relatively small. (mAP: 40.3 → 41.9)
- However, for the same settings, the mAP on unseen classes varies much more significantly (27.4 → 34.4).
- Mask head architectures play a critical role in generalization to unseen classes.

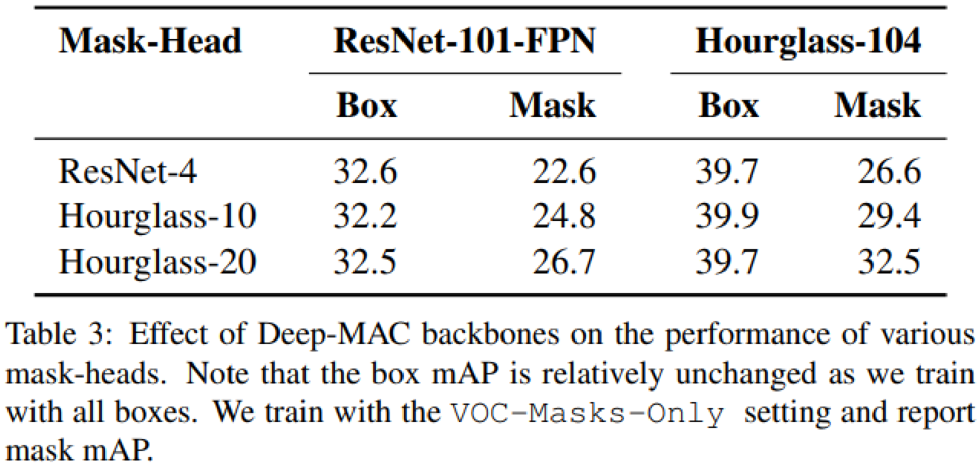


Mask-Head	**VOC mAP**		**Non-VOC mAP**	
	Prop. + GT.	**GT.**	**Prop. + GT.**	**GT.**
Default	40.9	41.1	19.6	29.3
ResNet-4	39.2	40.3	21.0	27.4
HG-20	41.6	41.4	20.6	33.8
HG-52	42.0	41.9	20.6	**34.4**

Table 2: Performance of Deep-MARC with different mask-heads under the `VOC-Masks-Only` setting with a ResNet-101-FPN backbone, comparing the performance when training the proposed boxes and groundtruth boxes (**Prop.+GT.**) and only groundtruth boxes (**GT.**). We see that performance on unseen classes depends significantly on the mask-head, but the benefit of better mask-heads is only apparent when training with groundtruth boxes. With the Hourglass (HG-52) mask-head and no other bells or whistles, Deep-MARC surpasses the previous state-of-the-art [10].

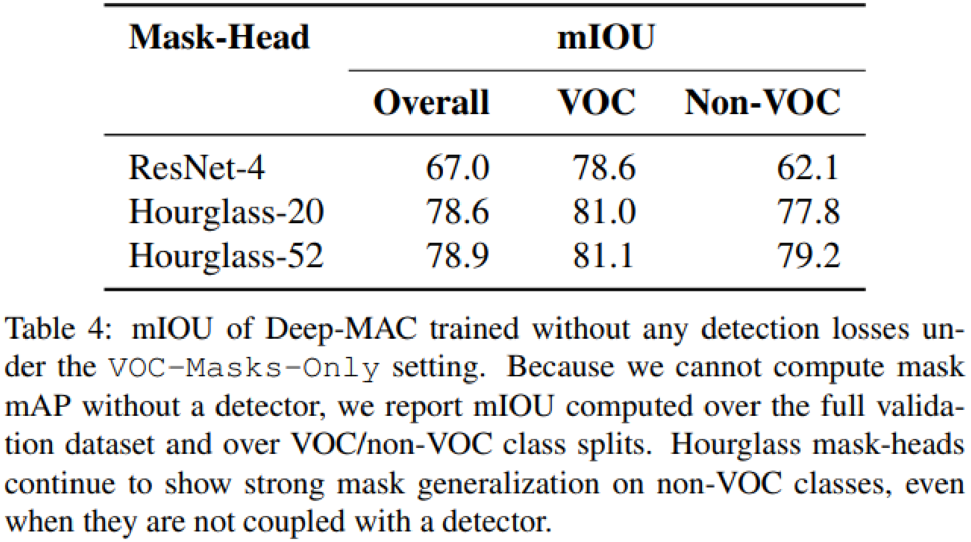
Going Deeper with Mask Heads

- Deep-MAC
- Strong mask generalization without a detector
  - Drop all detection related loss and evaluate using the mean IoU



Mask-Head	ResNet-101-FPN		Hourglass-104	
	Box	Mask	Box	Mask
ResNet-4	32.6	22.6	39.7	26.6
Hourglass-10	32.2	24.8	39.9	29.4
Hourglass-20	32.5	26.7	39.7	32.5

Table 3: Effect of Deep-MAC backbones on the performance of various mask-heads. Note that the box mAP is relatively unchanged as we train with all boxes. We train with the `VOC-Masks-Only` setting and report mask mAP.



Mask-Head	mIOU		
	Overall	VOC	Non-VOC
ResNet-4	67.0	78.6	62.1
Hourglass-20	78.6	81.0	77.8
Hourglass-52	78.9	81.1	79.2

Table 4: mIOU of Deep-MAC trained without any detection losses under the `VOC-Masks-Only` setting. Because we cannot compute mask mAP without a detector, we report mIOU computed over the full validation dataset and over VOC/non-VOC class splits. Hourglass mask-heads continue to show strong mask generalization on non-VOC classes, even when they are not coupled with a detector.

Going Deeper with Mask Heads

- Deep-MAC
- From a classical perspective, this is counterintuitive given the over-parameterization of very deep mask-heads,
- but perhaps is not so surprising in light of recent ways of rethinking generalization for deep learning. (Chiyuan Zhang et al.)

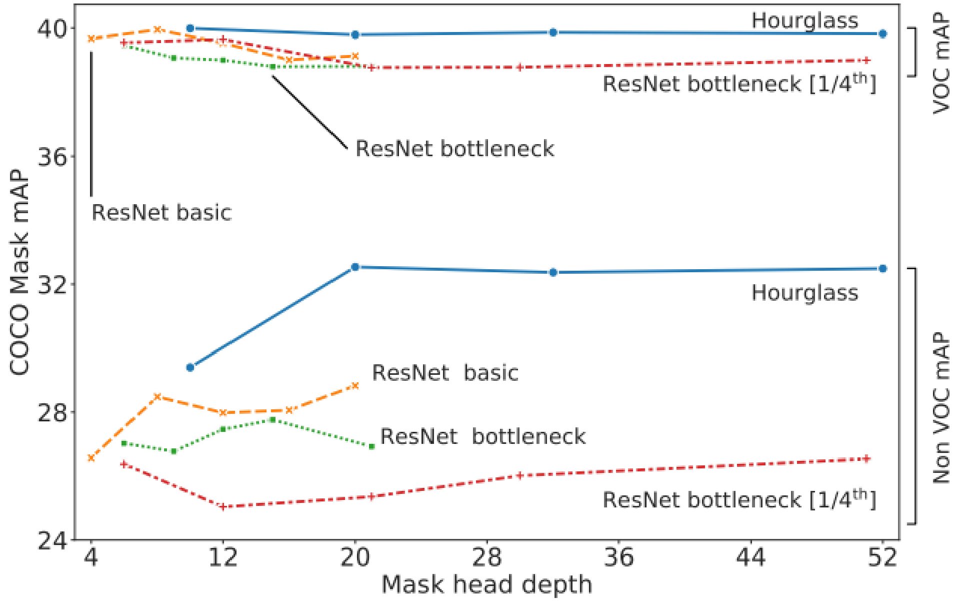


Figure 3: Effect of mask-head architecture and depth on instance segmentation performance over seen (VOC) and unseen (Non-VOC) classes. Although the performance on seen classes does not vary much across different architectures, there is significant variation in the performance on unseen classes. We report results with the `VOC-Masks-Only` setting.

Understanding deep learning requires rethinking generalization, Chiyuan Zhang et al., https://arxiv.org/abs/1611.03530
Identity crisis: Memorization and generalization under extreme overparameterization, Chiyuan Zhang et al., https://arxiv.org/abs/1902.04698

A Closer Look at Mask-head Architecture

- Hourglass Networks

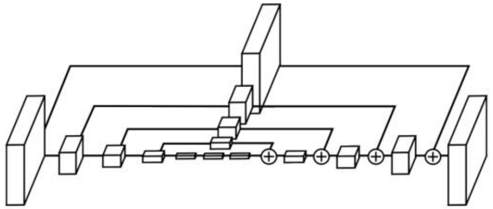


Fig. 3. An illustration of a single "hourglass" module. Each box in the figure corresponds to a residual module as seen in Figure 4. The number of features is consistent across the whole hourglass.

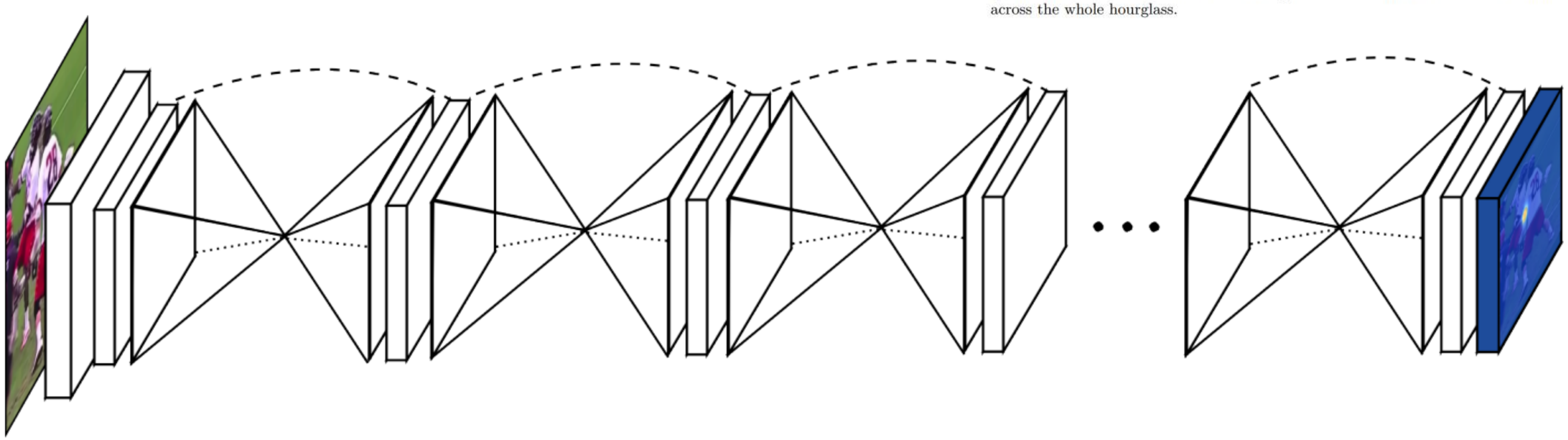


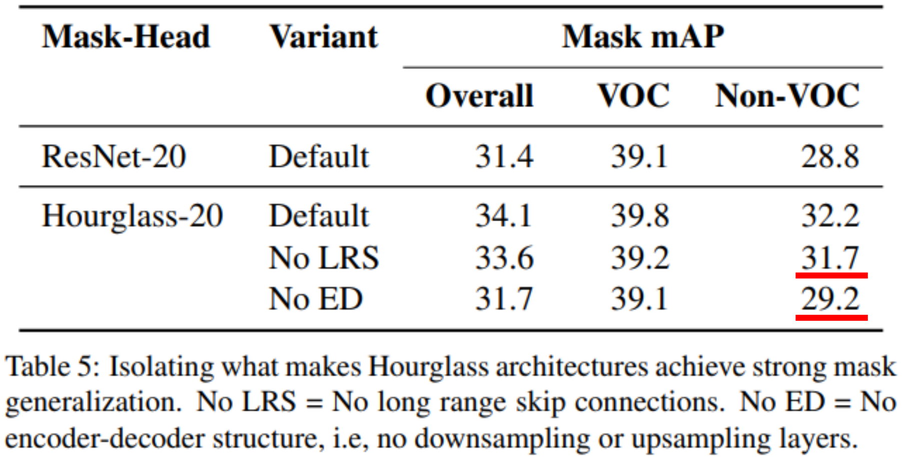
Fig. 1. Our network for pose estimation consists of multiple stacked hourglass modules which allow for repeated bottom-up, top-down inference.

Stacked Hourglass Networks for Human Pose Estimation, Alejandro Newell et al., https://arxiv.org/abs/1603.06937

A Closer Look at Mask-head Architecture

- What makes Hourglass mask-heads so good?

1. Encoder-decoder structure (Down-sampling/up-sampling)
   - Encoder-decoder structure is appropriate inductive biases for segmentation
2. Long range skip connection



Mask-Head	**Variant**	**Mask mAP**		
		Overall	**VOC**	**Non-VOC**
ResNet-20	Default	31.4	39.1	28.8
Hourglass-20	Default	34.1	39.8	32.2
	No LRS	33.6	39.2	31.7
	No ED	31.7	39.1	29.2

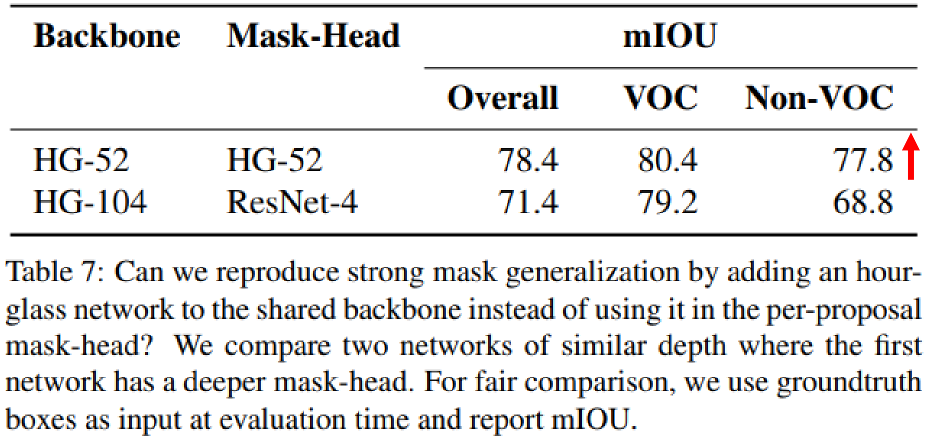
Win!

Table 5: Isolating what makes Hourglass architectures achieve strong mask generalization. No LRS = No long range skip connections. No ED = No encoder-decoder structure, i.e, no downsampling or upsampling layers.

Stacked Hourglass Networks for Human Pose Estimation, Alejandro Newell et al., https://arxiv.org/abs/1603.06937

A Closer Look at Mask-head Architecture

- What’s so special about the mask-head?
- Reproduce advantages by adding an HG network to the shared backbone instead of using it in the per-proposal mask-head
  - HG-52 + HG-52 vs. HG-104 + ResNet-4
- Mask-head plays a significant role with respect to generalization to unseen classes

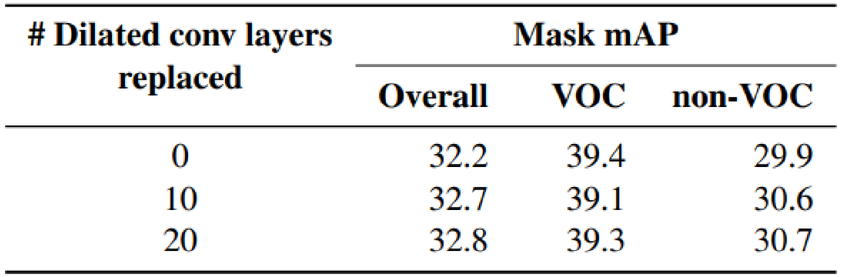


Backbone	**Mask-Head**	**mIOU**		
		Overall	**VOC**	**Non-VOC**
HG-52	HG-52	78.4	80.4	77.8 ↑
HG-104	ResNet-4	71.4	79.2	68.8

Table 7: Can we reproduce strong mask generalization by adding an hourglass network to the shared backbone instead of using it in the per-proposal mask-head? We compare two networks of similar depth where the first network has a deeper mask-head. For fair comparison, we use groundtruth boxes as input at evaluation time and report mIOU.

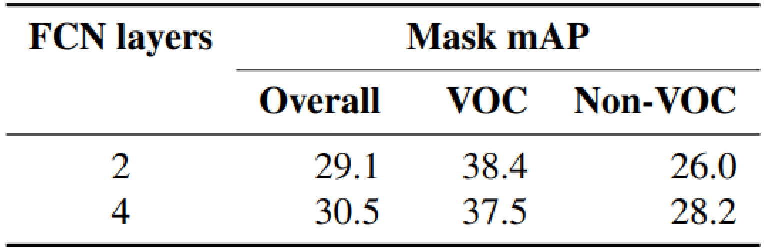
A Closer Look at Mask-head Architecture

- Is it sufficient to have a large receptive field?
  - Dilated convolutional mask-head
  - Fully connected mask-head



# Dilated conv layers replaced	Mask mAP		
	Overall	VOC	non-VOC
0	32.2	39.4	29.9
10	32.7	39.1	30.6
20	32.8	39.3	30.7

Table 6: Replacing regular convolutional layers with dilated convolutions (rate=2) in a ResNet-20 mask-head to isolate the effect of receptive field.

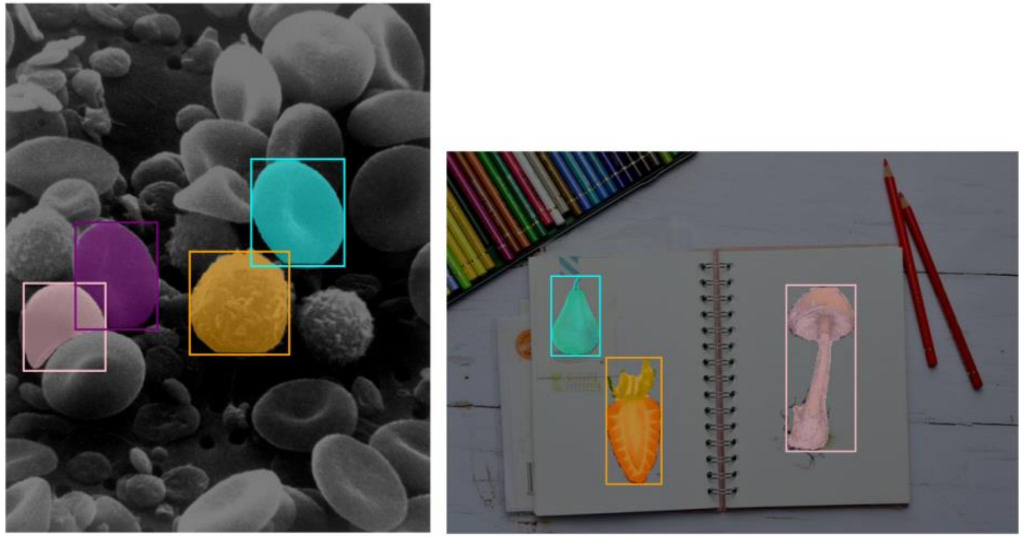


FCN layers	Mask mAP		
	Overall	VOC	Non-VOC
2	29.1	38.4	26.0
4	30.5	37.5	28.2

Table 11: Effect of using fully connected layers as mask-heads on Deep-MAC . Performance is reported with the `VOC-Masks-Only` setup. For easy reference, the VOC/non-VOC mask mAP values for Resnet-4 and HG-52 mask-heads are 39.7/26.6 and 39.8/32.5 respectively.

Cropping to Only Ground Truth Boxes

- A ground truth box, when tight on an instance, acts as a cue, indicating the object that is meant to be segmented.
- When trained on noisy proposals, we conjecture that Mask R-CNN tries to memorize the types of foreground classes seen at training time and thus struggles to generalize to unseen classes.
- Requires a large enough receptive field so that boundary pixels can interact with interior pixels.



(e) SEM blood cells image from wikipedia.

(f) Photo by Maggie Jaszowska on Unsplash.

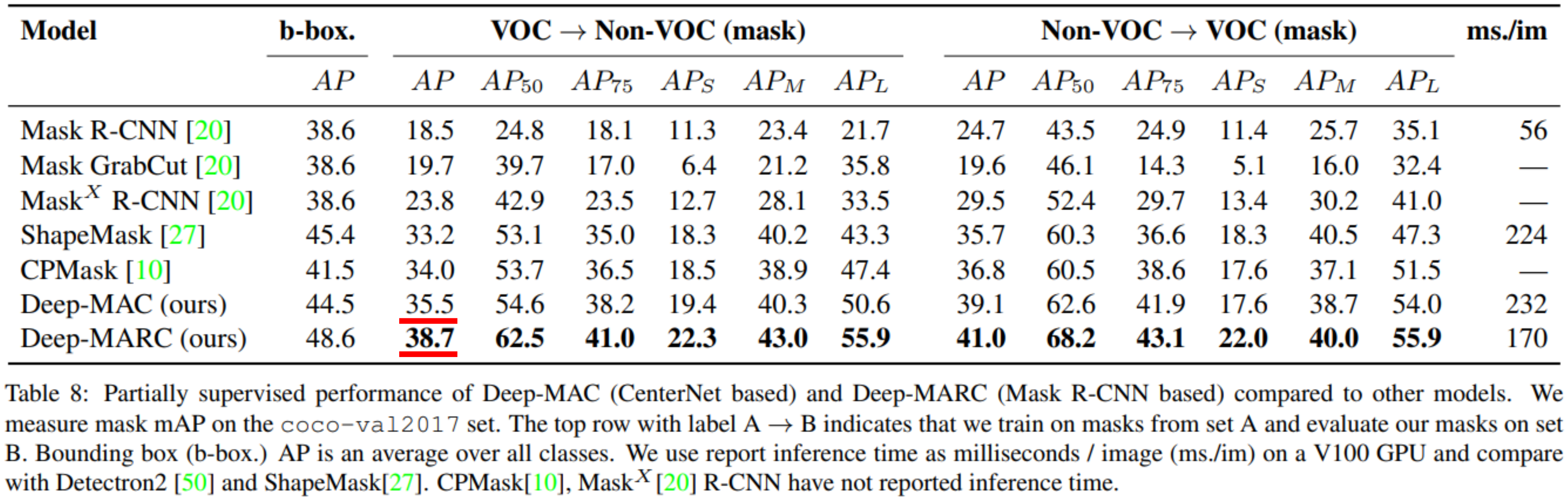
Figure 6: Example outputs of Deep-MAC with hand-drawn boxes on unknown classes.

https://github.com/google/deepmac/blob/main/assets/deepmac_video.mp4?raw=true

Comparison with the state-of-the-art

- Deep-MAC uses an Hourglass-104 backbone and an Hourglass-100 mask-head
- Deep-MARC uses a SpineNet-143 backbone and an Hourglass-52 mask-head.

Fully supervised setup
39.4 mAP
42.8 mAP



Model	b-box.	VOC → Non-VOC (mask)						Non-VOC → VOC (mask)						ms./im
	AP	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	
Mask R-CNN [20]	38.6	18.5	24.8	18.1	11.3	23.4	21.7	24.7	43.5	24.9	11.4	25.7	35.1	56
Mask GrabCut [20]	38.6	19.7	39.7	17.0	6.4	21.2	35.8	19.6	46.1	14.3	5.1	16.0	32.4	—
Mask^X R-CNN [20]	38.6	23.8	42.9	23.5	12.7	28.1	33.5	29.5	52.4	29.7	13.4	30.2	41.0	—
ShapeMask [27]	45.4	33.2	53.1	35.0	18.3	40.2	43.3	35.7	60.3	36.6	18.3	40.5	47.3	224
CPMask [10]	41.5	34.0	53.7	36.5	18.5	38.9	47.4	36.8	60.5	38.6	17.6	37.1	51.5	—
Deep-MAC (ours)	44.5	35.5	54.6	38.2	19.4	40.3	50.6	39.1	62.6	41.9	17.6	38.7	54.0	232
Deep-MARC (ours)	48.6	**38.7**	**62.5**	**41.0**	**22.3**	**43.0**	**55.9**	**41.0**	**68.2**	**43.1**	**22.0**	**40.0**	**55.9**	170

Table 8: Partially supervised performance of Deep-MAC (CenterNet based) and Deep-MARC (Mask R-CNN based) compared to other models. We measure mask mAP on the `coco-val2017` set. The top row with label A → B indicates that we train on masks from set A and evaluate our masks on set B. Bounding box (b-box.) AP is an average over all classes. We use report inference time as milliseconds / image (ms./im) on a V100 GPU and compare with Detectron2 [50] and ShapeMask[27]. CPMask[10], Mask^X [20] R-CNN have not reported inference time.

References

- Learning to Segment Every Thing, Ronghang Hu et al., https://arxiv.org/abs/1711.10370
- Extreme clicking for efficient object annotation, Dim P. Papadopoulos et al., https://arxiv.org/abs/1708.02750
- ShapeMask: Learning to Segment Novel Objects by Refining Shape Priors, Weicheng Kuo et al., https://arxiv.org/abs/1904.03239
- Mask R-CNN, Kaiming He et al., https://arxiv.org/abs/1703.06870
- FCOS: Fully Convolutional One-Stage Object Detection, Zhi Tian et al., https://arxiv.org/abs/1904.01355
- Objects as Points, Xingyi Zhou et al., https://arxiv.org/abs/1904.07850
- Stacked Hourglass Networks for Human Pose Estimation, Alejandro Newell et al., https://arxiv.org/abs/1603.06937