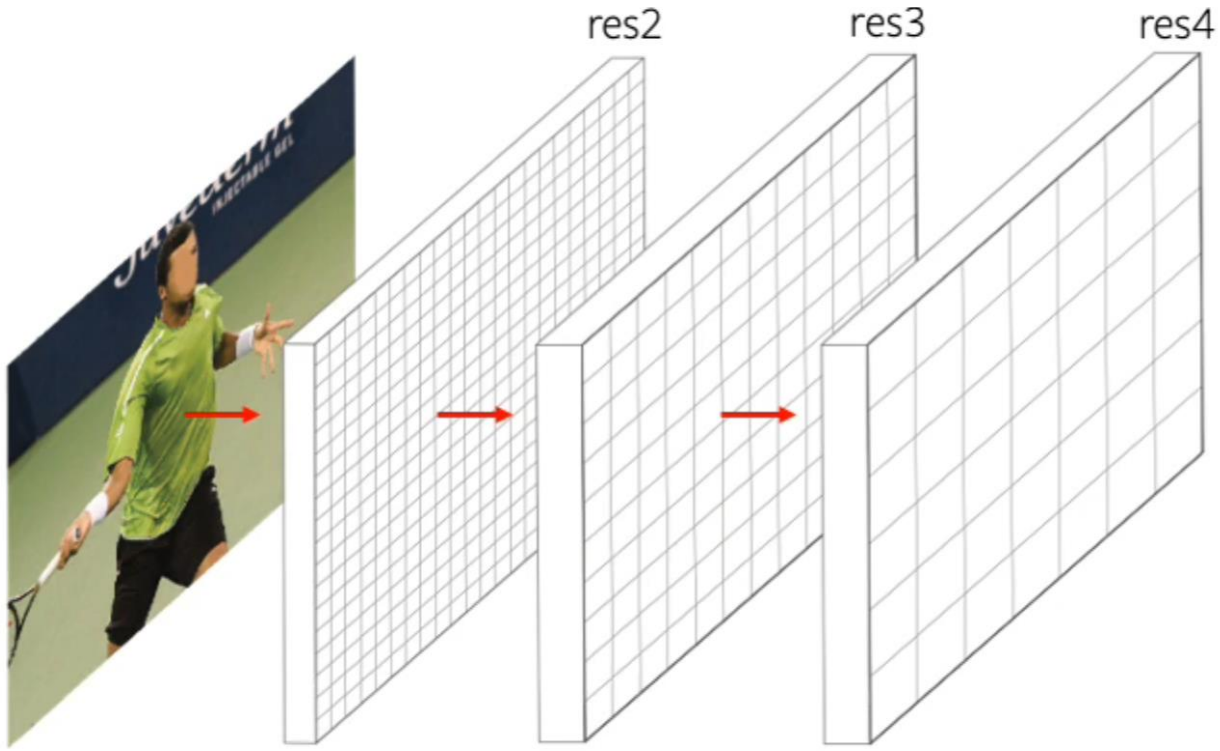
PointRend

Image Segmentation as Rendering (CVPR 2020)



Su Hyung Choi

Korea University

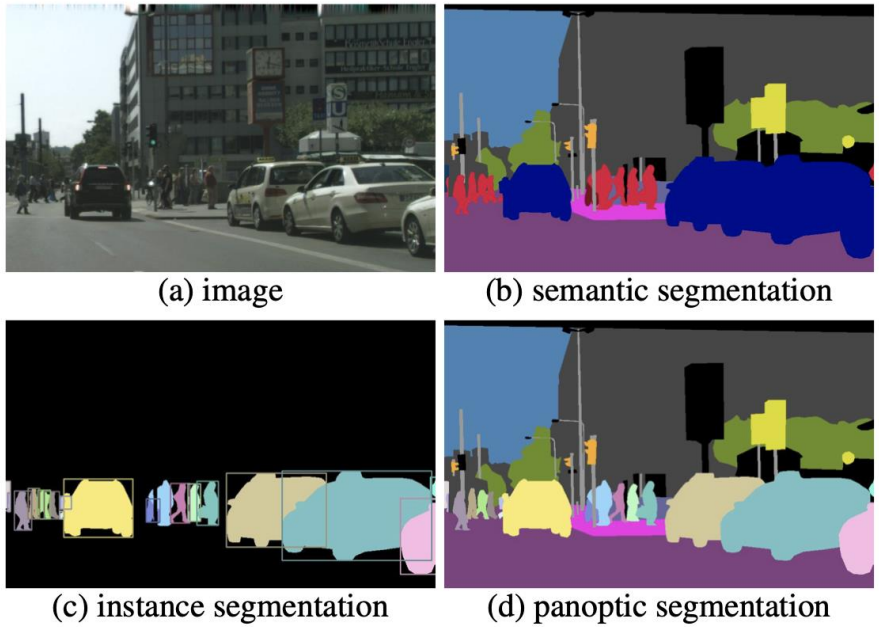
Research Intern @ Computer Vision Lab

Submitted on Jan 11, 2022

Authors: Facebook AI Research (FAIR)

Alexander Kirillov, Yuxin Wu, Kaiming He Ross Girshick

Background



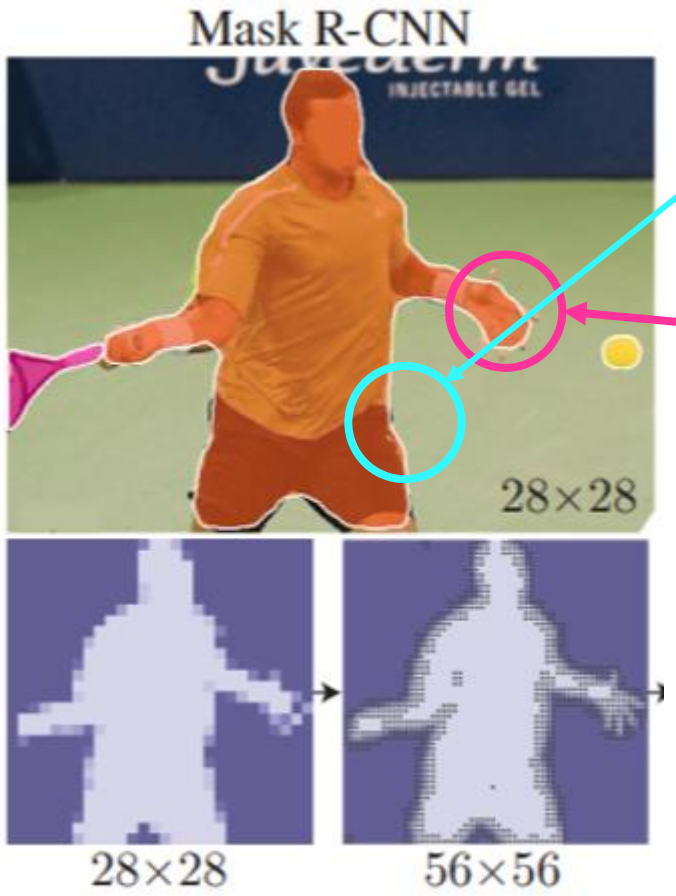
"Enables having a **global view** of image segmentation, both **category-wise** as well as **instance-wise**. Hence the name '**PANOPTIC**' , the means showing or seeing everything at once."

"Panoptic Segmentation"

Facebook AI Research (FAIR) - Leading by Kaiming He.

- Panoptic Segmentation (CVPR 2019)
- Panoptic Feature Pyramid Networks (CVPR 2019)
- PointRend: Image Segmentation as Rendering (CVPR 2020)

Typical Problems in Image Segmentation



Low Frequency

=> Sufficient Low Resolution

High Frequency

=> Require High Resolution

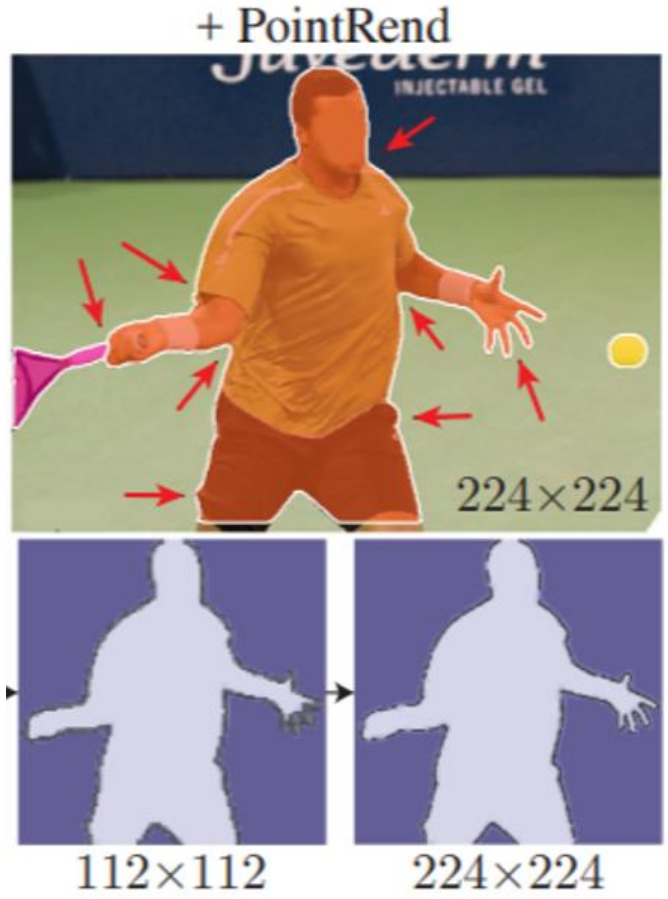
A 28×28 regular grid irrespective of object

=> Predicted label maps are mostly smooth

The regular grid unnecessarily oversample the smooth areas

The regular grid undersampling object boundaries

PointRend



Propose "Point based Rendering" as a methodology for image segmentation using point representations.

To Efficiently "render" high quality label map

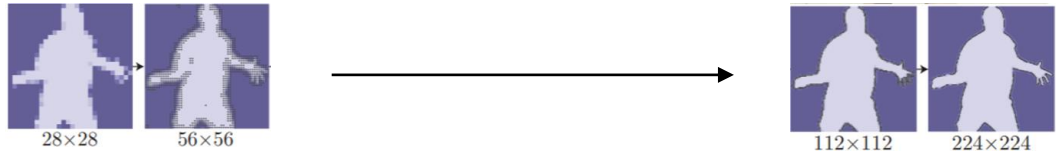
Benefits over other architectures:

1. Training and Inference is less computationally intensive
2. Generate higher quality, higher resolution label maps (masks)

PointRend

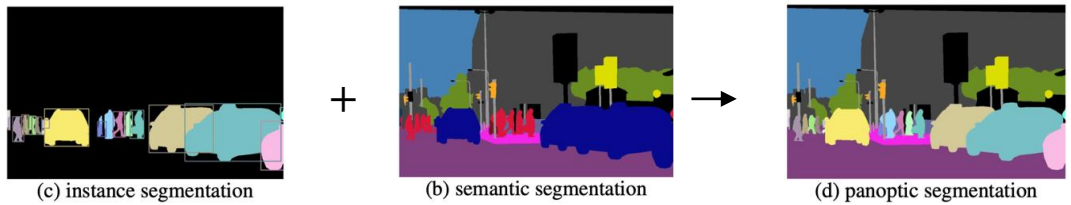
Three Points of PointRend

New method for efficient high-quality image segmentation of objects and scenes.



Qualitatively, PointRend outputs crisp object boundaries in regions that are oversmoothed by previous methods.

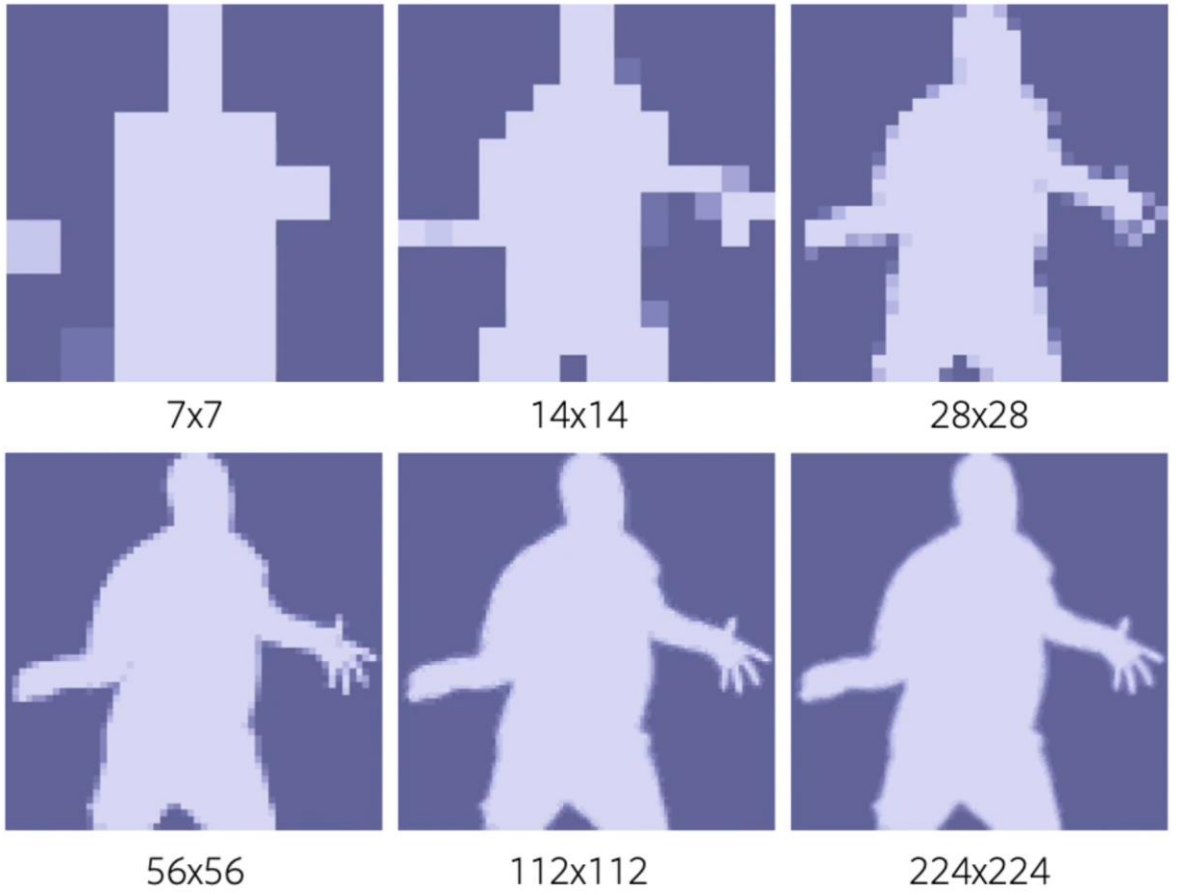
Quantitatively, PointRend yields significant gains on COCO and Cityscapes, for both instance and semantic segmentation.

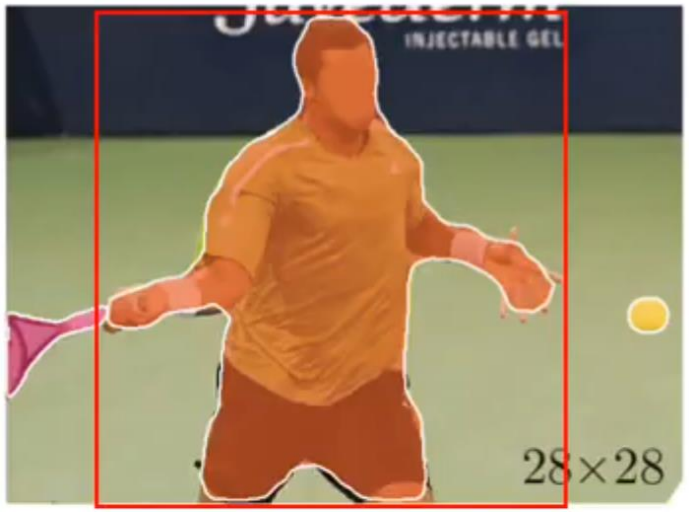


"**Unique perspective** of image segmentation as a rendering problem"

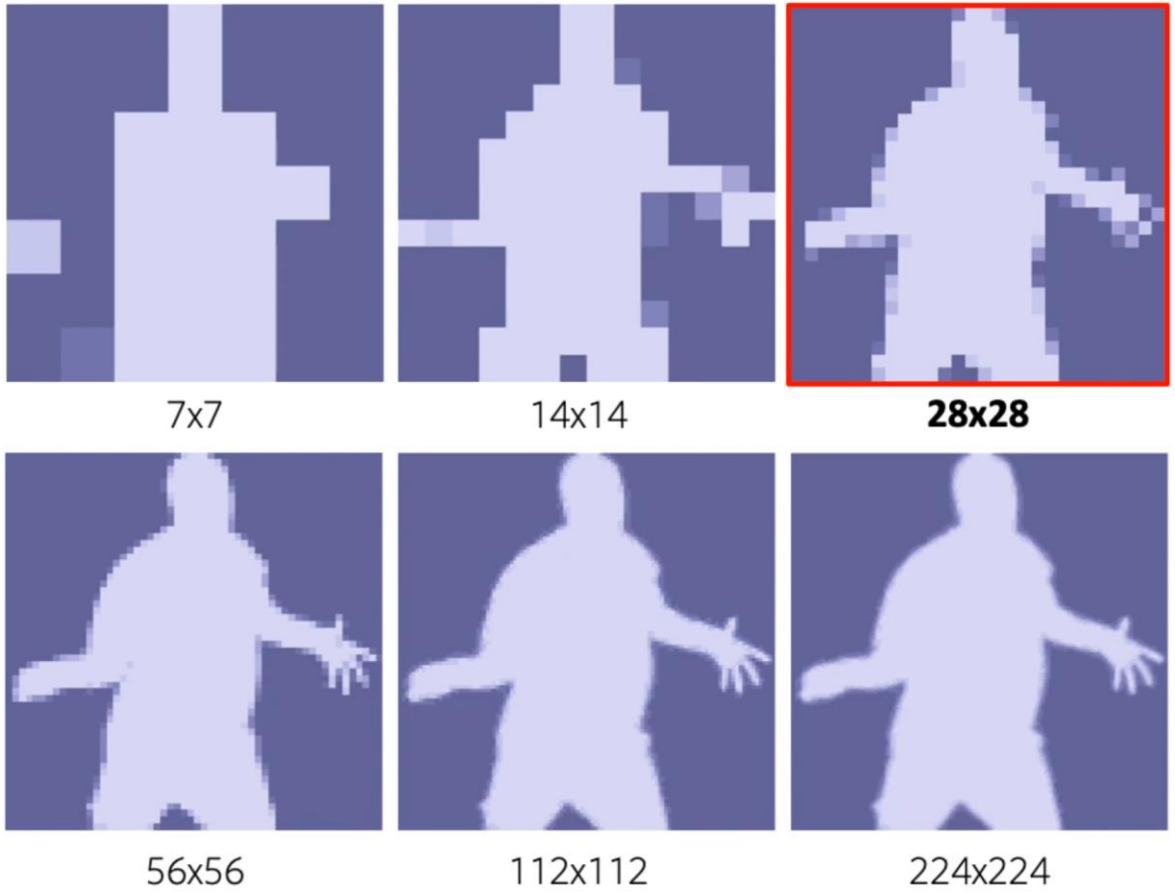
Explain

Output resolution is tradeoff between computational cost and level of detail

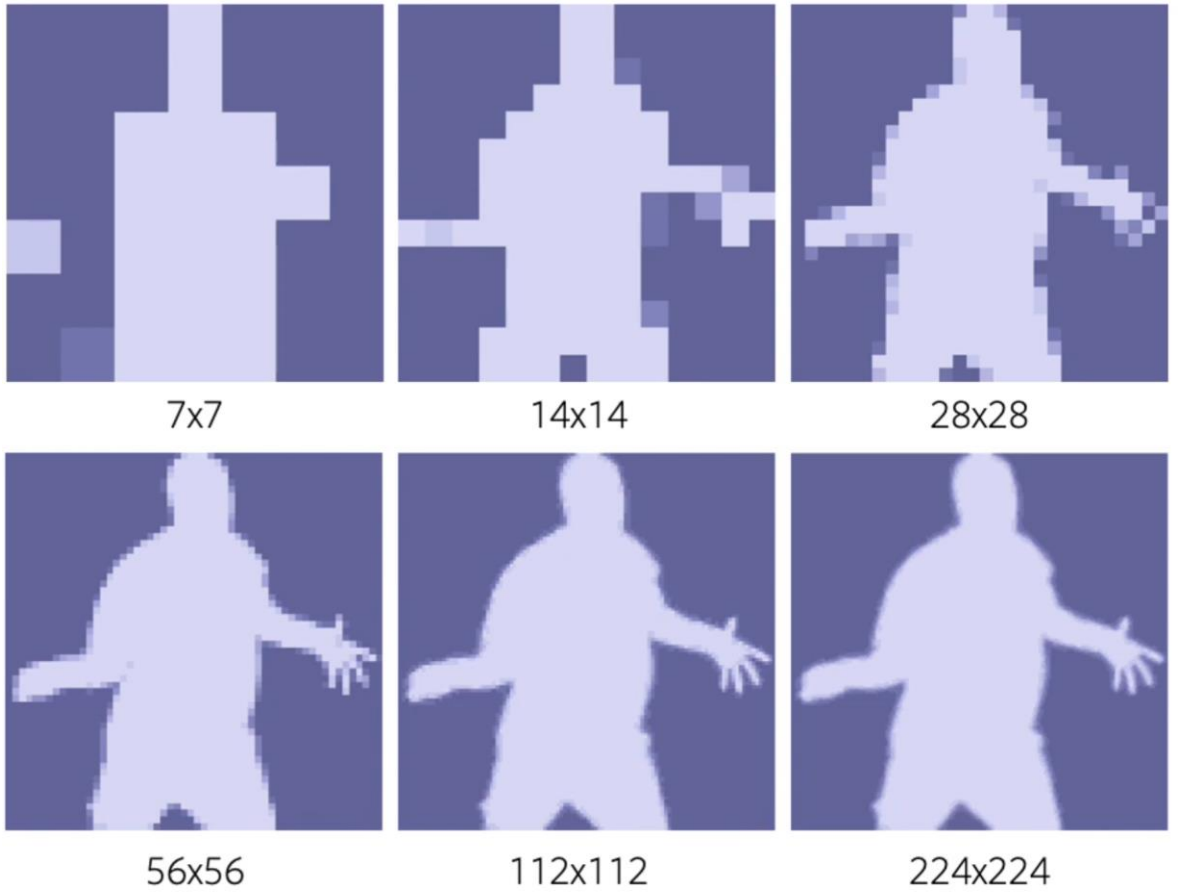




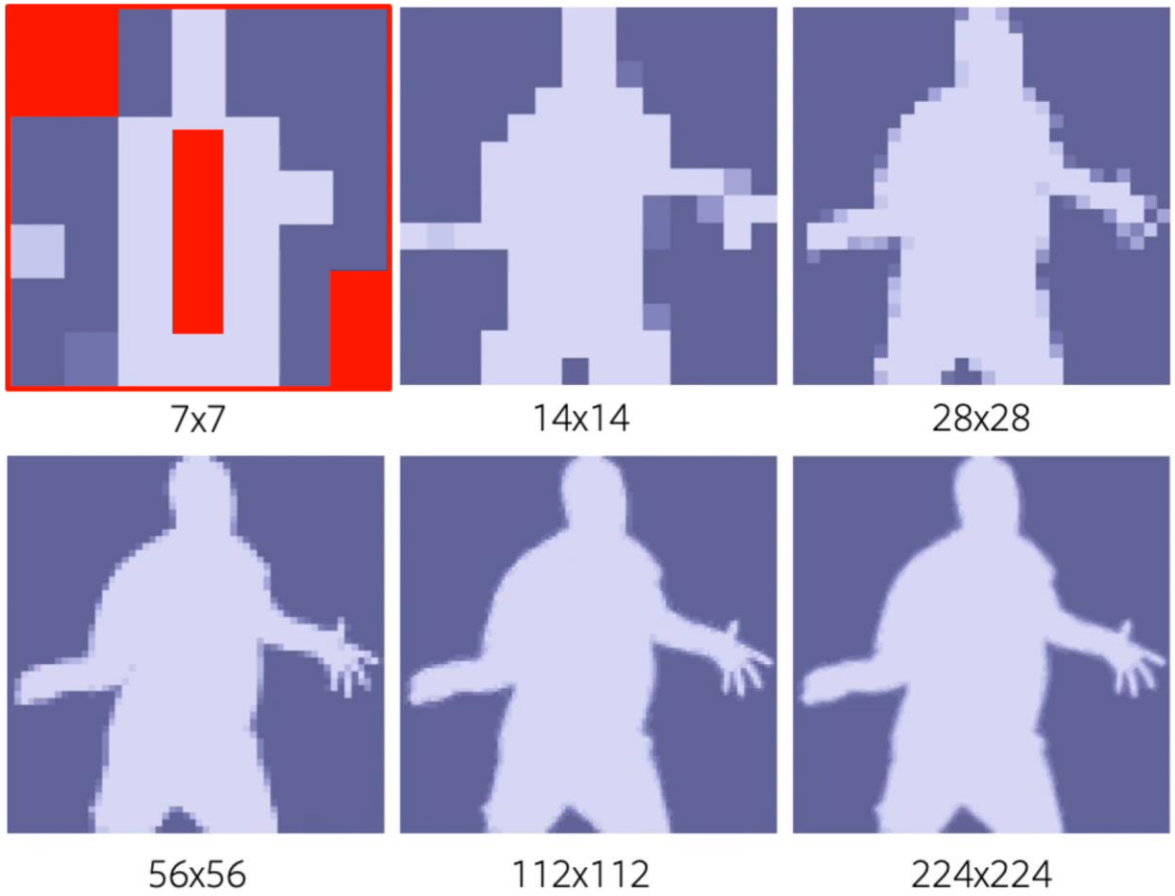
Mask R-CNN efficiently predicts low-resolution masks



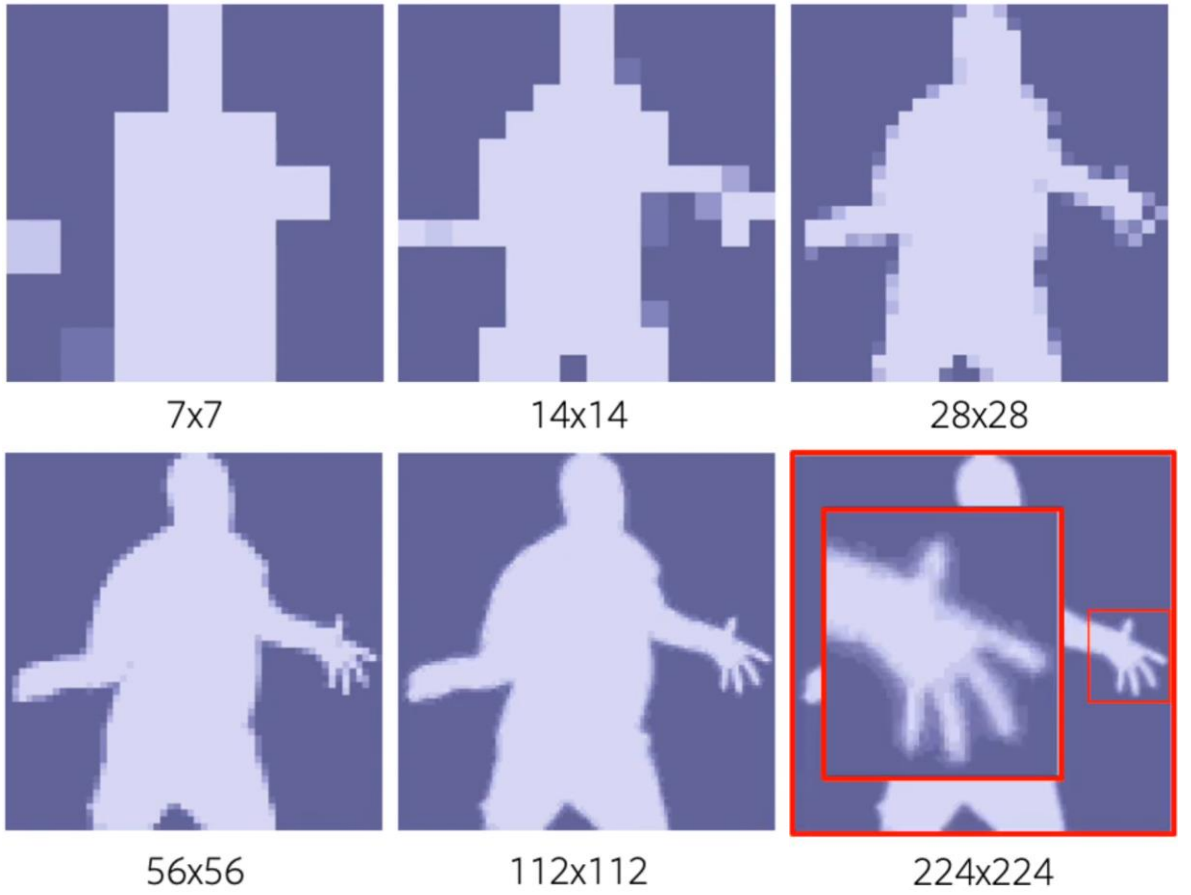
But, different areas require different levels of detail



Some are perfectly segmented with low resolution prediction



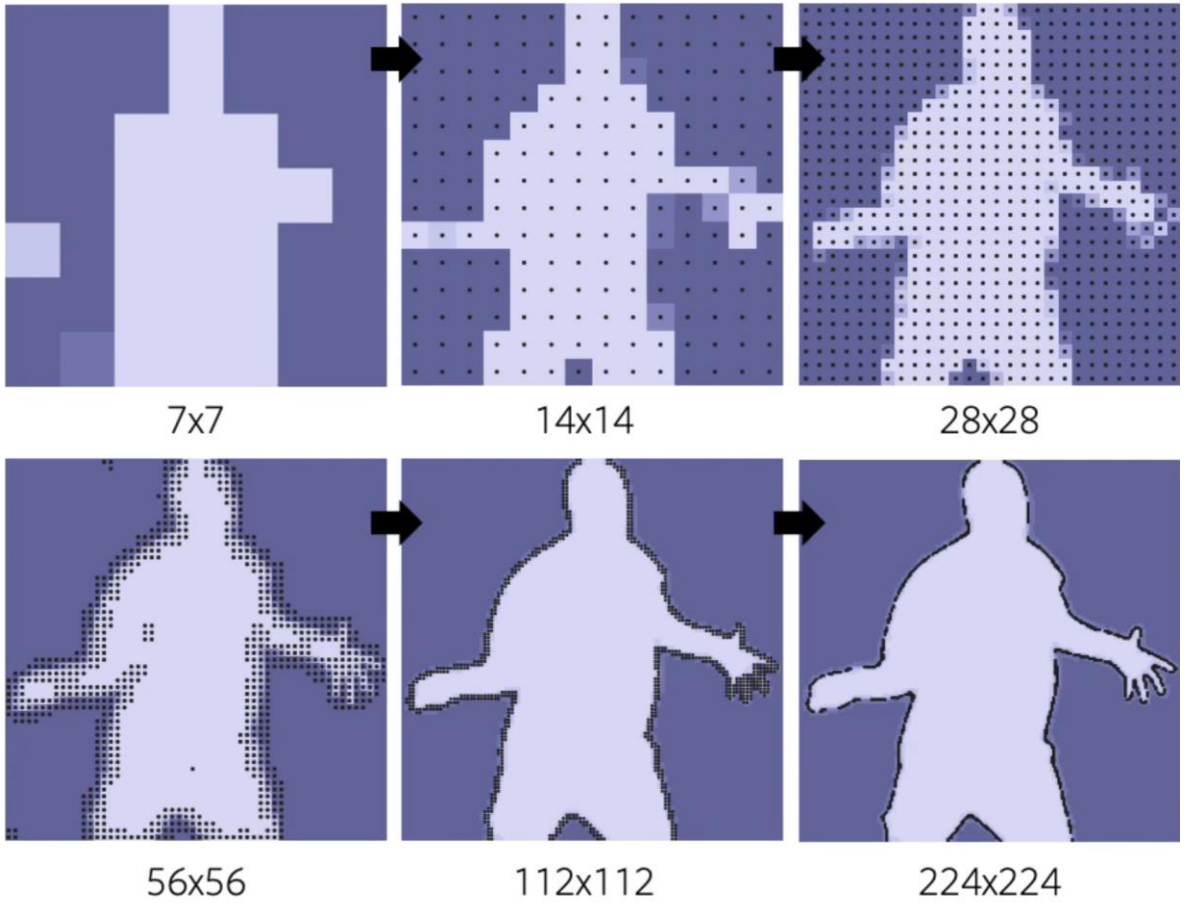
Whereas high-frequency regions require prediction in a very high resolution



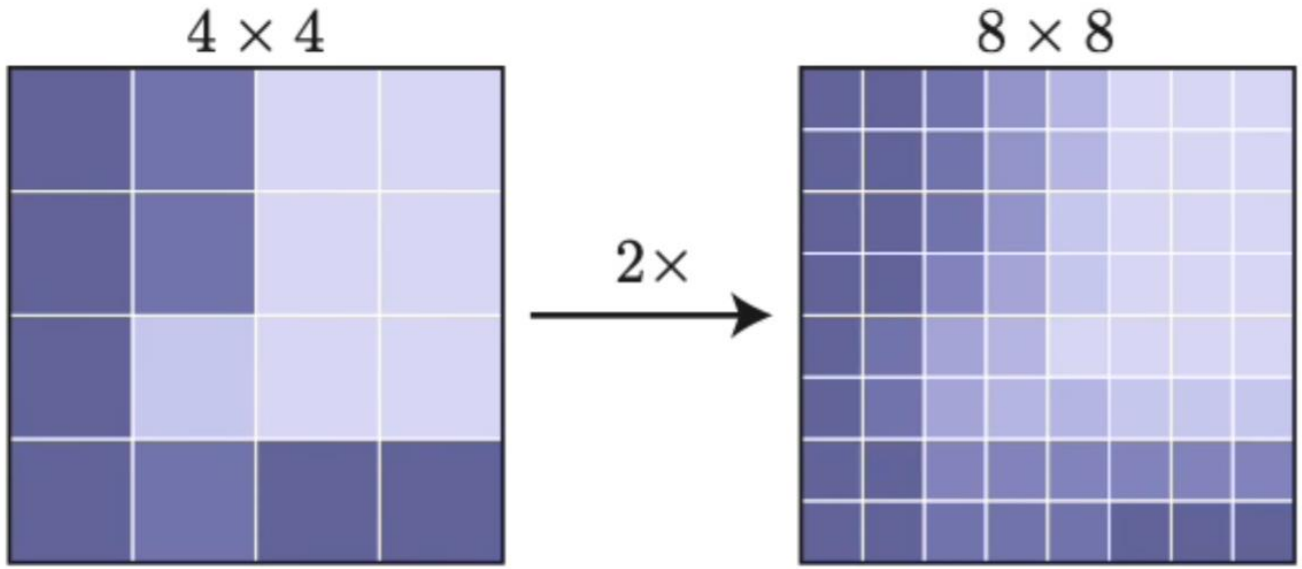
=> Require High resolution but computational and memory costs should be too high.

Solution

PointRend gradually increases resolution by making predictions for the most uncertain points

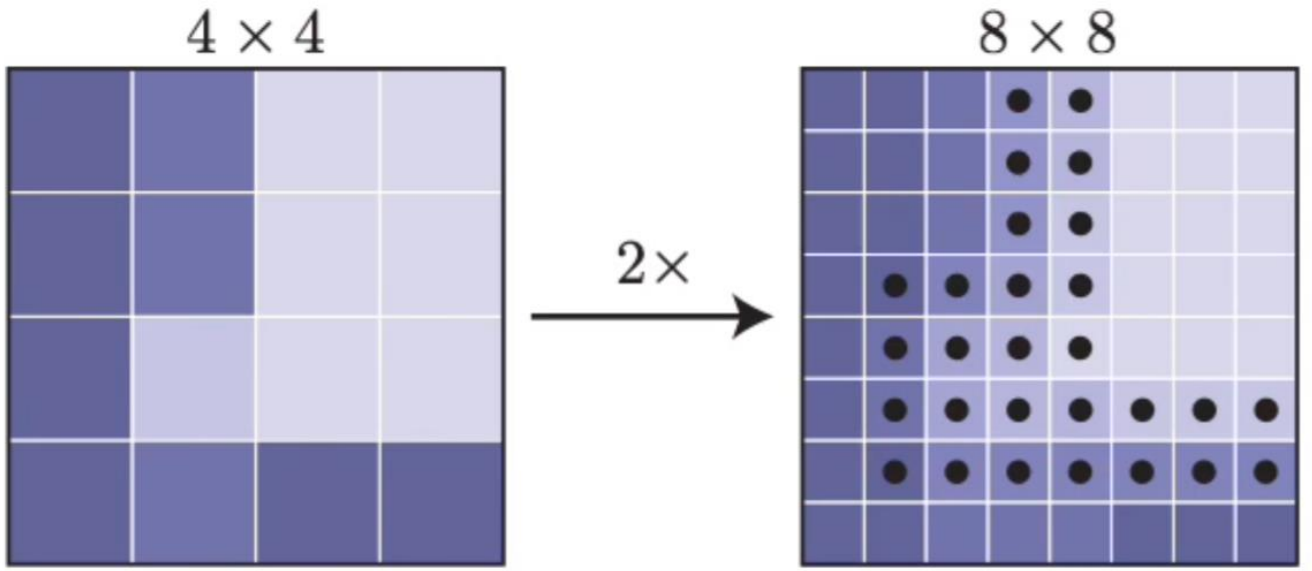


Point-based inference via subdivision



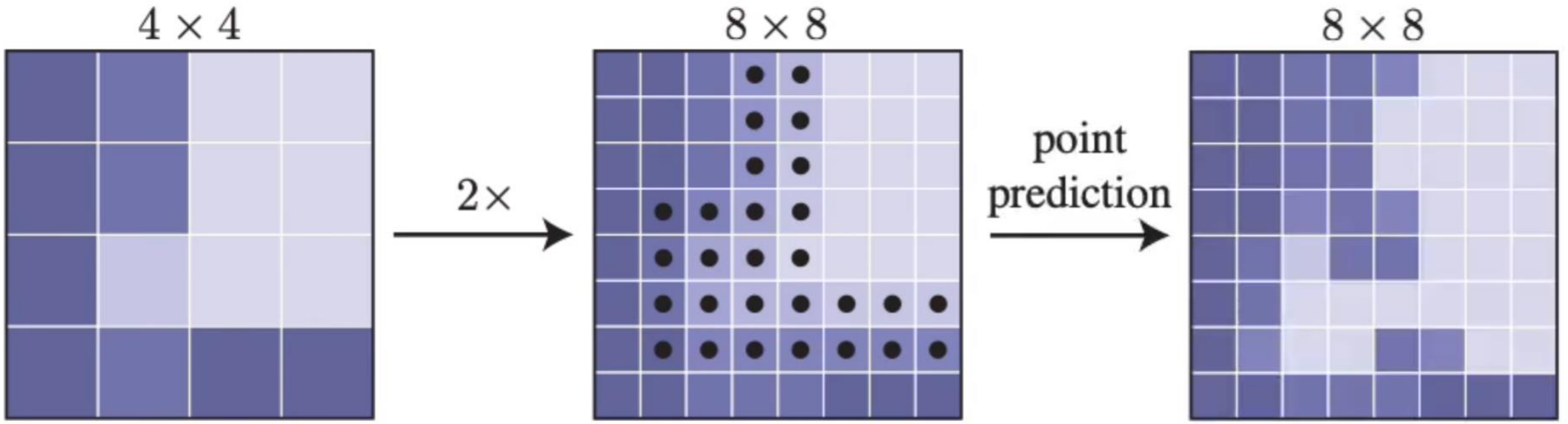
Lower resolutions prediction is
unsampled with bilinear interpolation

Point-based inference via subdivision



The subset of most uncertain points is selected

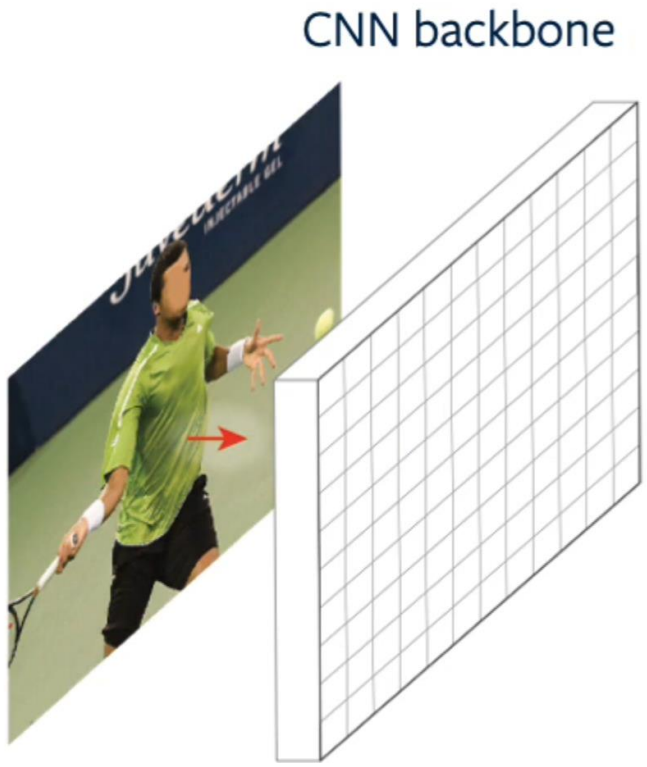
Point-based inference via subdivision



Prediction for each selected points is refined using a lightweight MLP

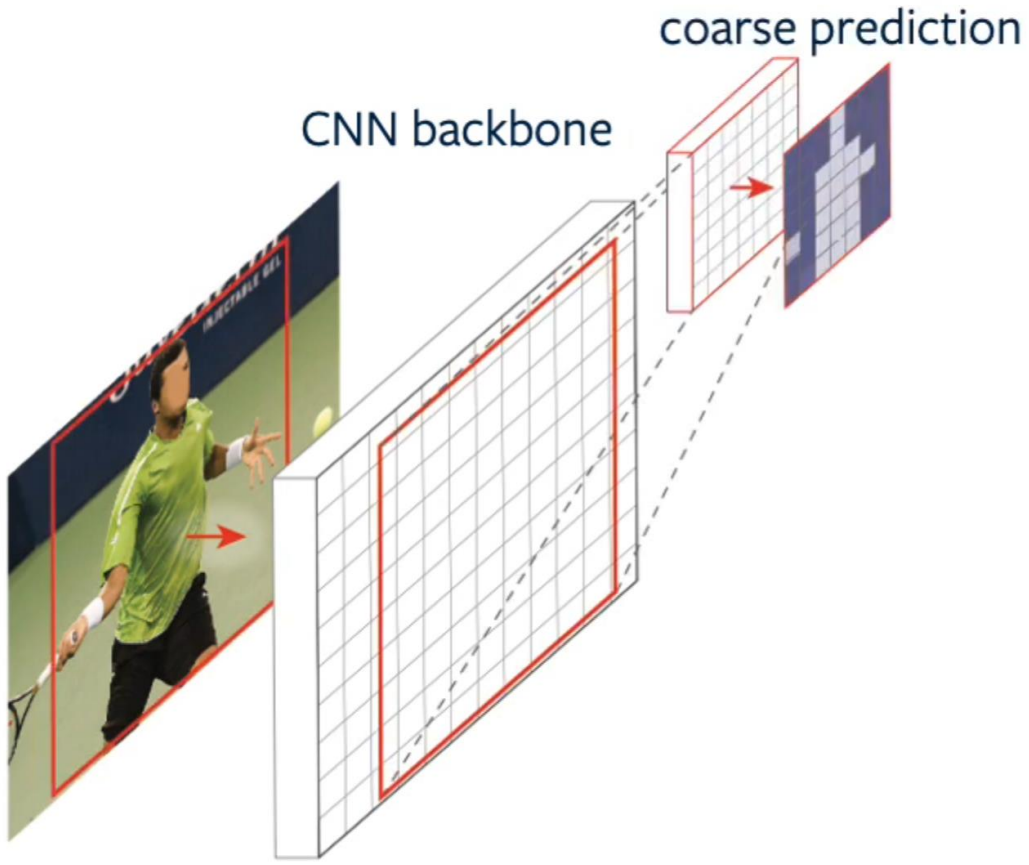
Explain overall Architecture

PointRend architecture for instance segmentation



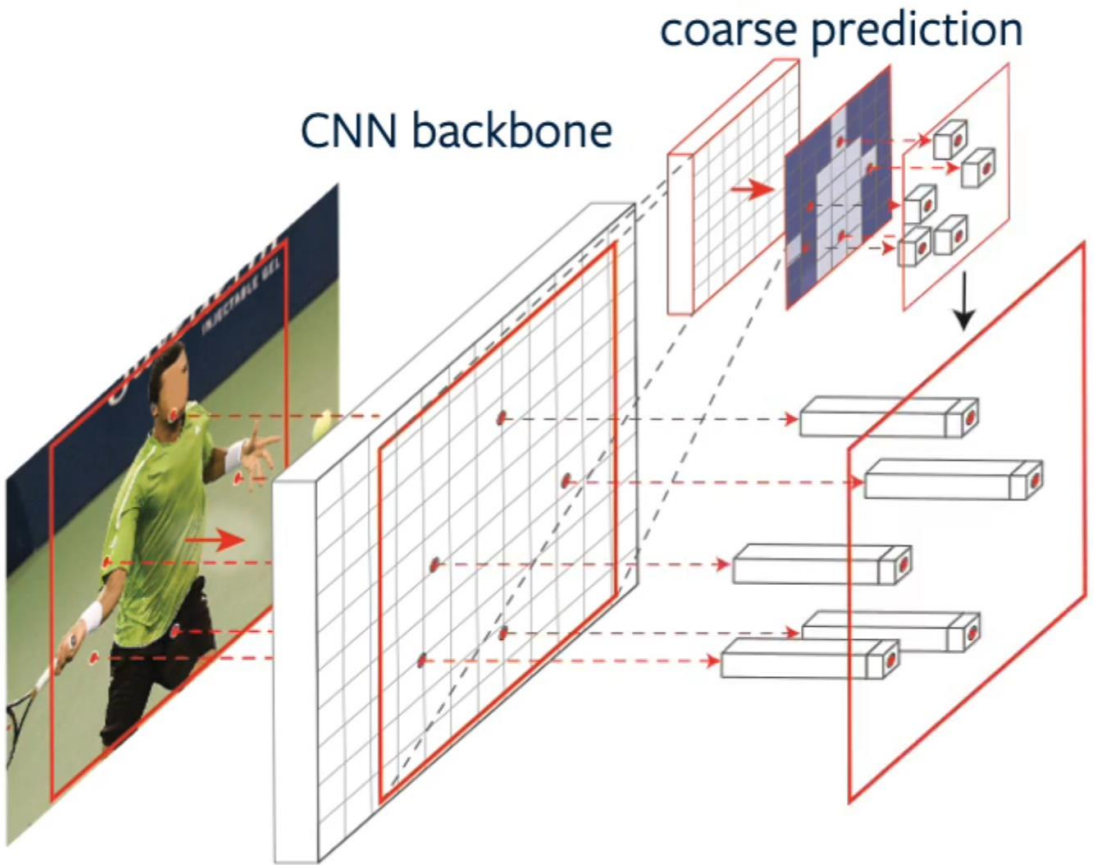
Backbone computes features
that represent the whole image

PointRend architecture for instance segmentation



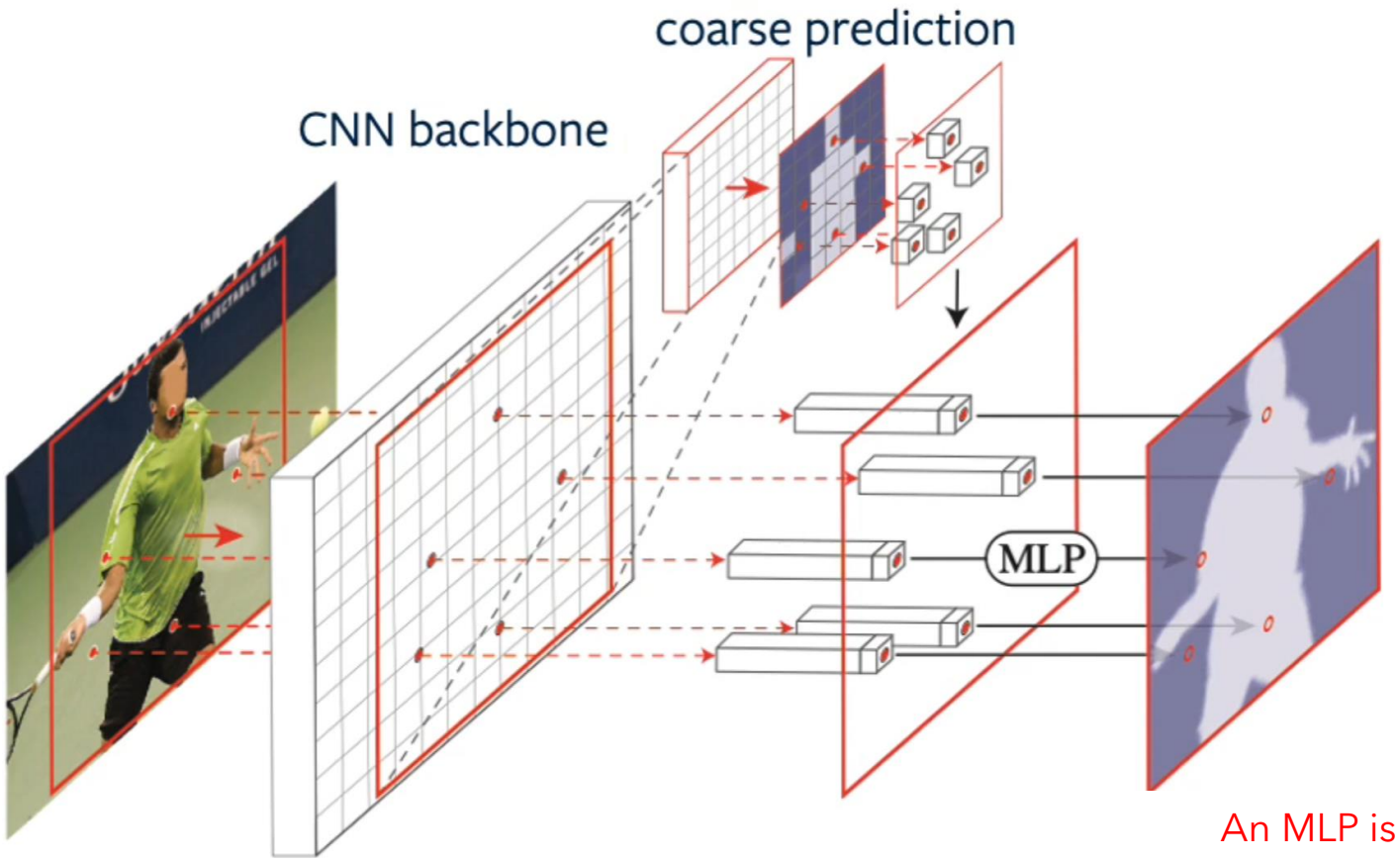
For each bounding box
a small head yields
low-resolution mask prediction

PointRend architecture for instance segmentation



For a subset of points we extract features from the coarse prediction and the backbone features using bilinear interpolation

PointRend architecture for instance segmentation

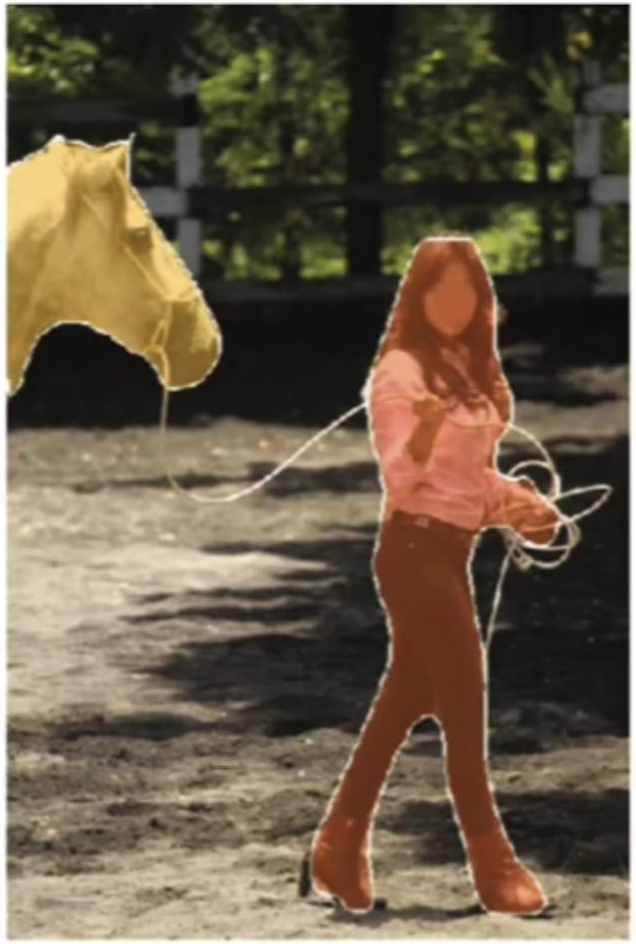


An MLP is used to make prediction for each point independently

Result for instance segmentation

Mask R-CNN with standard head vs. Mask R-CNN with PointRend

+ PointRend



+ PointRend

Quantitative comparison: instance segmentation

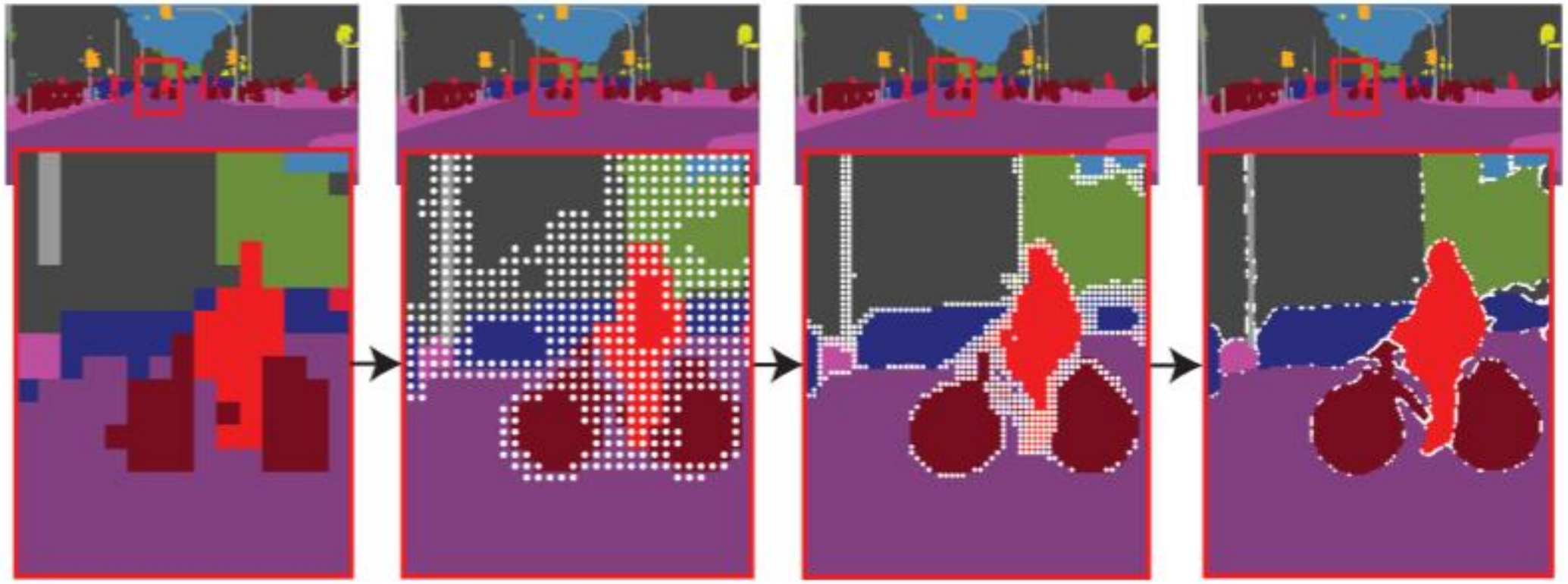
mask head	output resolution	COCO AP	COCO AP*	Cityscapes AP
4× conv	28×28	35.2	37.6	33.0
PointRend	224×224	**36.3** (+1.1)	**39.7** (+2.1)	**35.8** (+2.8)

Mask R-CNN with standard head (4x Conv) vs. Mask R-CNN with PointRend

AP* is COCO mask AP for a COCO-trained model
evaluated against the higher quality LVIS annotation

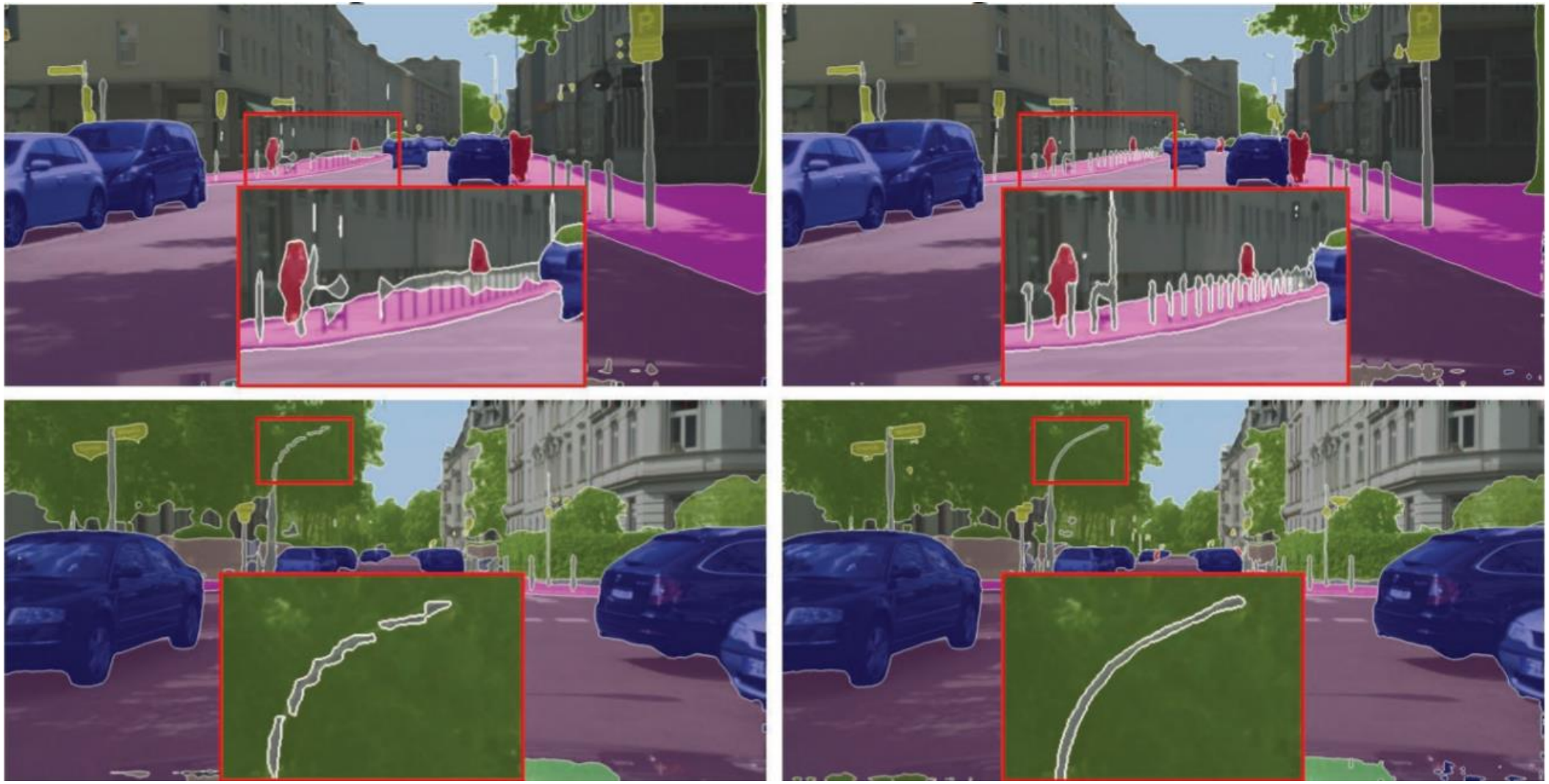
Result for semantic segmentation

PointRend for semantic segmentation



PointRend can be applied on top of
any modern semantic segmentation model

Deeplab V3 vs. Deeplab V3 + PointRend



+ PointRend

Quantitative comparison: semantic segmentation

method	output resolution	mIoU
DeeplabV3-OS-16	64×128	77.2
DeeplabV3-OS-8	128×256	77.8 (+0.6)
DeeplabV3-OS-16 + PointRend	1024×2048	**78.4** **(+1.2)**

DeeplabV3 vs. DeeplabV3 with PointRend

method	output resolution	mIoU
SemanticFPN P_2-P_5	256×512	77.7
SemanticFPN P_2-P_5 + PointRend	1024×2048	**78.6** **(+0.9)**
SemanticFPN P_3-P_5	128×256	77.4
SemanticFPN P_3-P_5 + PointRend	1024×2048	**78.5** **(+1.1)**

SemanticFPN vs SemanticFPN with PointRend

Conclusion

- **High resolution output with little to no computational overhead**

  Higher resolution, more accurate masks

  with fewer model params, less compute time.

- **“Plug & play”** on top of any FCN-based model for segmentation
- **Significant quantitative and qualitative improvement**

Thank You.

[Paper Review] PointRend: Image Segmentation as Rendering

Su Hyung Choi