ECCV'20 ONLINE
23-28 AUGUST 2020

# Dense RepPoints: Representing Visual Objects with Dense Point Sets

Ze Yang [1]*, Yinghao Xu [2]*, Han Xue [3]*, Zheng Zhang [5],
Raquel Urtasun [4], Liwei Wang [1], Stephen Lin [5], Han Hu [5]

1 PEKING UNIVERSITY 1898

2

3 SHANGHAI JIAO TONG UNIVERSITY 1896

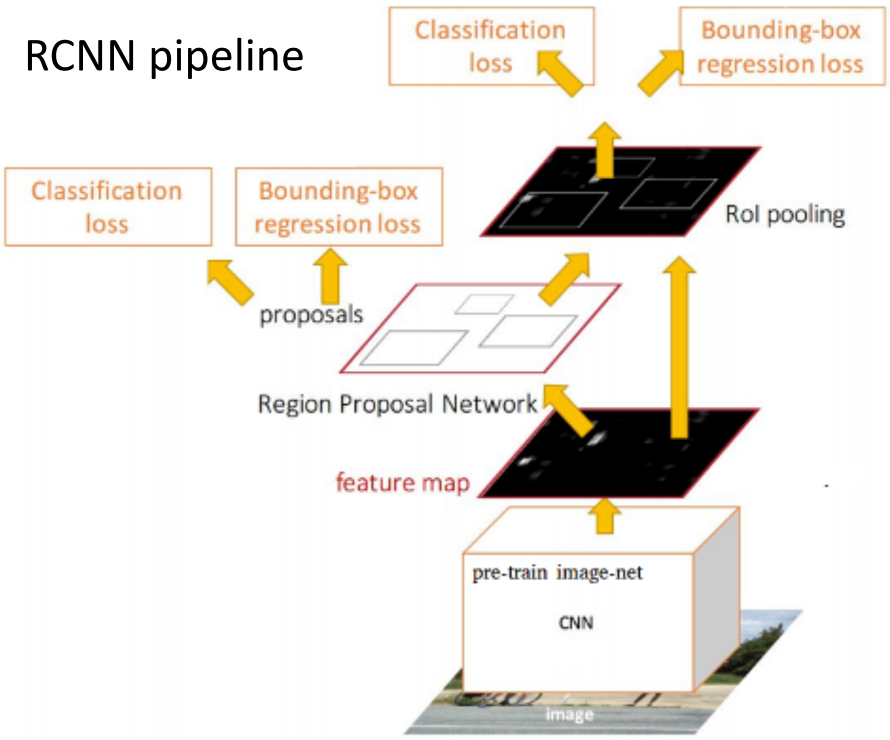
4 UNIVERSITY OF TORONTO

5 Microsoft Research 微软亚洲研究院

# Background

Current framework for visual perception system.

RCNN pipeline




$$\text{bbox anchors} \xrightarrow{\text{bbox reg.}} \text{bbox proposals (S1)}$$
$$\xrightarrow{\text{bbox reg.}} \text{bbox proposals (S2)}$$
$$\ldots$$
$$\xrightarrow{\text{bbox reg.}} \text{bbox object targets}$$

*Use bounding box as intermediate representation*

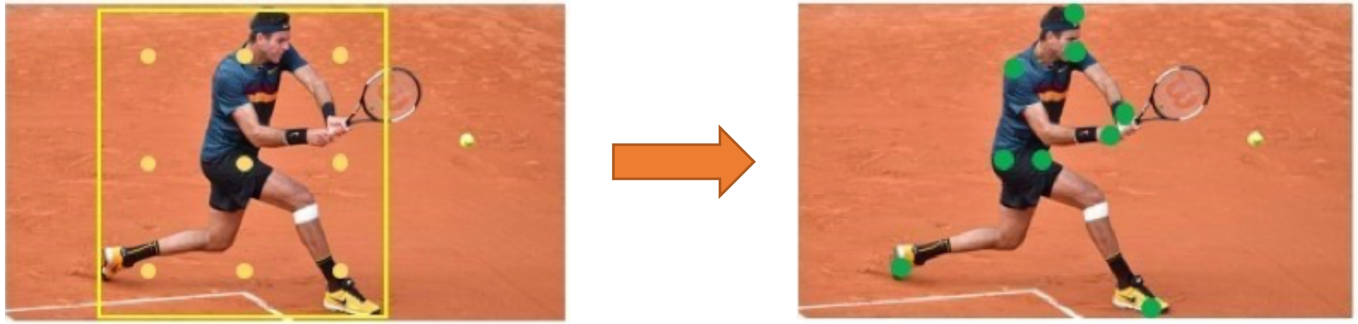
# Background

***Why bounding box?***

- Bounding box is convenient to annotate with little ambiguity.
- Easy feature extraction.

***Limitations.***

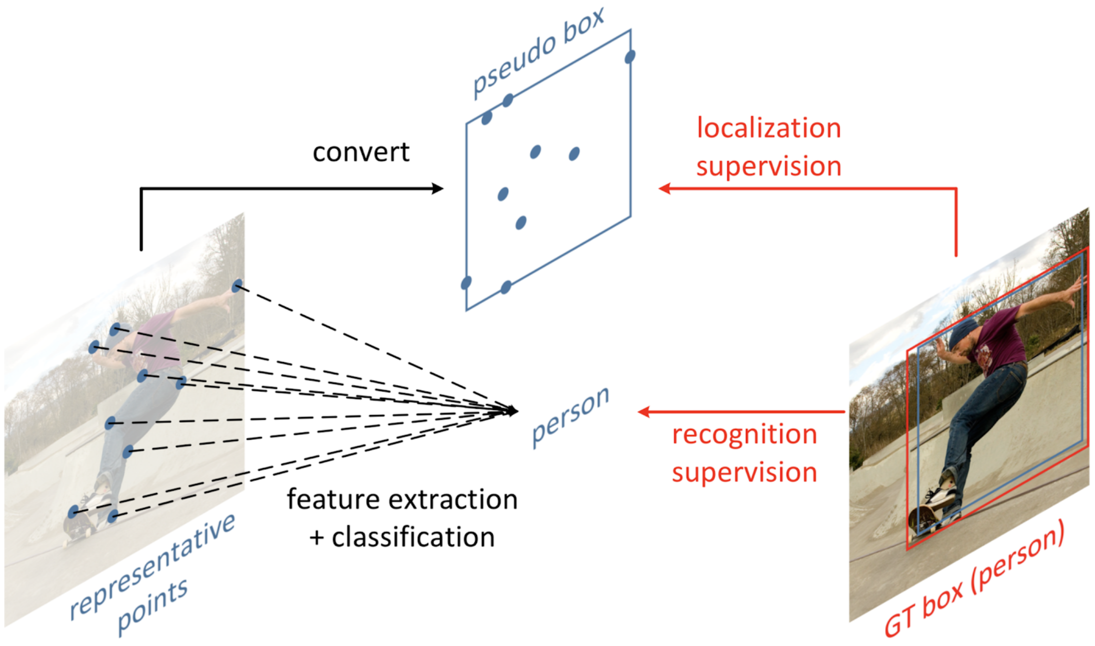
- Coarse object feature extraction.
- Unable to tackle irregular object, e.g. roads.



*Better geometric/semantic aligned representation for recognition?*

# Background

***RepPoints*** (representative points) for object detection




***RepPoints*** is a set of points connecting stages. It serves as:

1) **flexible geometric 2D representation**
2) **semantically aligned feature extraction.**

SoTA detector

# Background

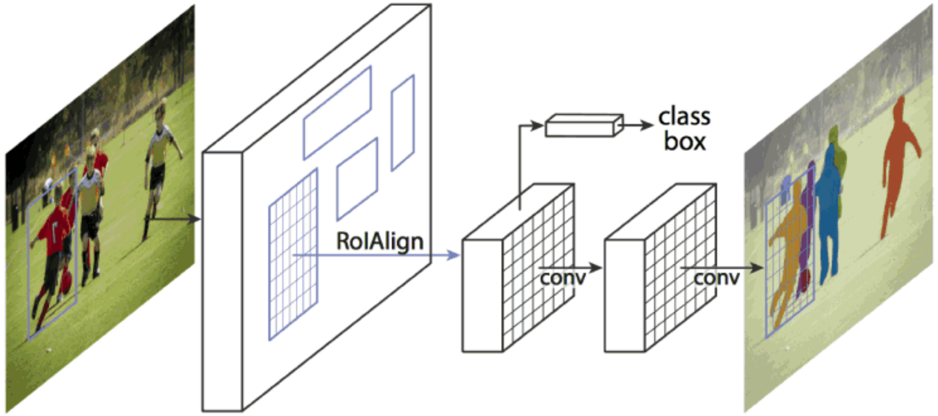
Can we extend representative points to dense segmentation tasks?

Contour

Foreground mask

# Instance segmentation representation

Foreground Mask Representation

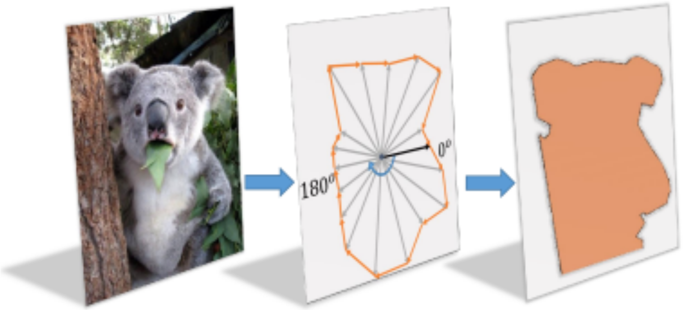



*RCNN framework*

1. Detect rectangular regions
2. Pixel-wise verification inside rectangular regions

Contour Representation

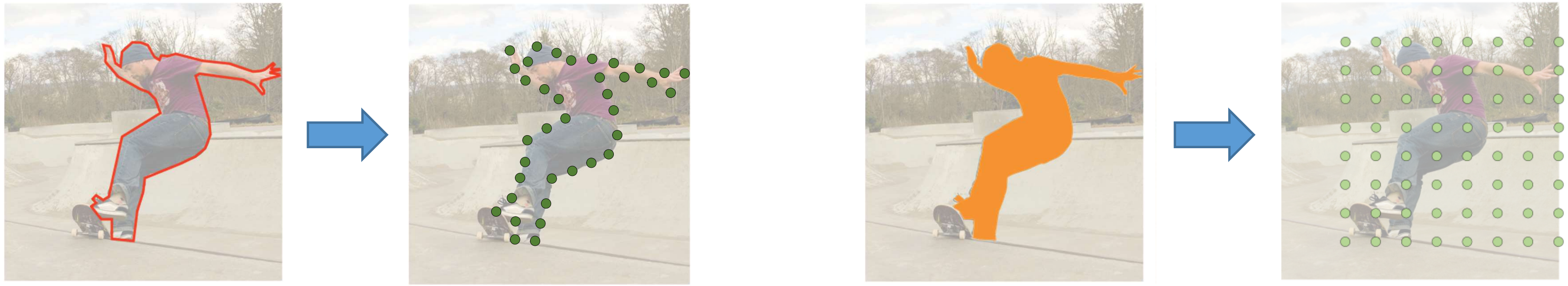
*Energy minimization framework*




*Learning contour regression*

# Dense RepPoints

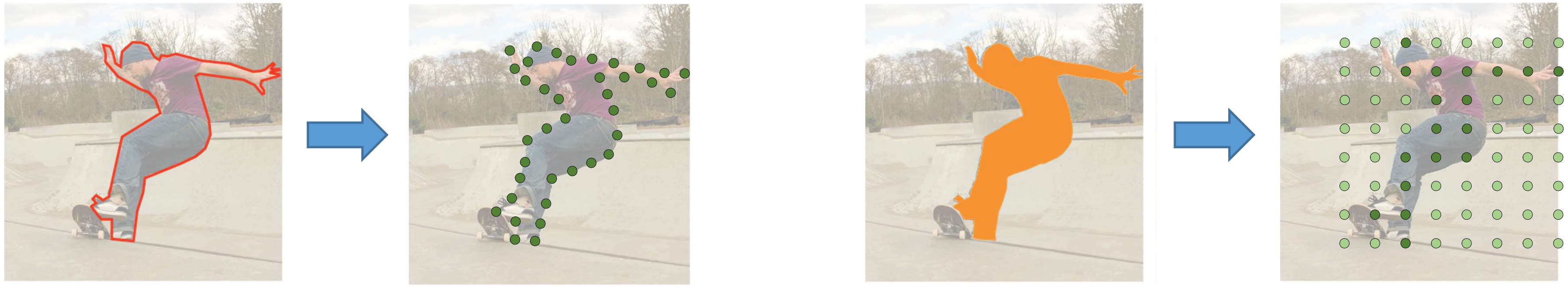
Use ***Dense RepPoints*** to represent contour and grid mask through sampling



contour | boundary sample | foreground mask | grid sample

# Dense RepPoints

Use ***Dense RepPoints*** to represent contour and grid mask through sampling



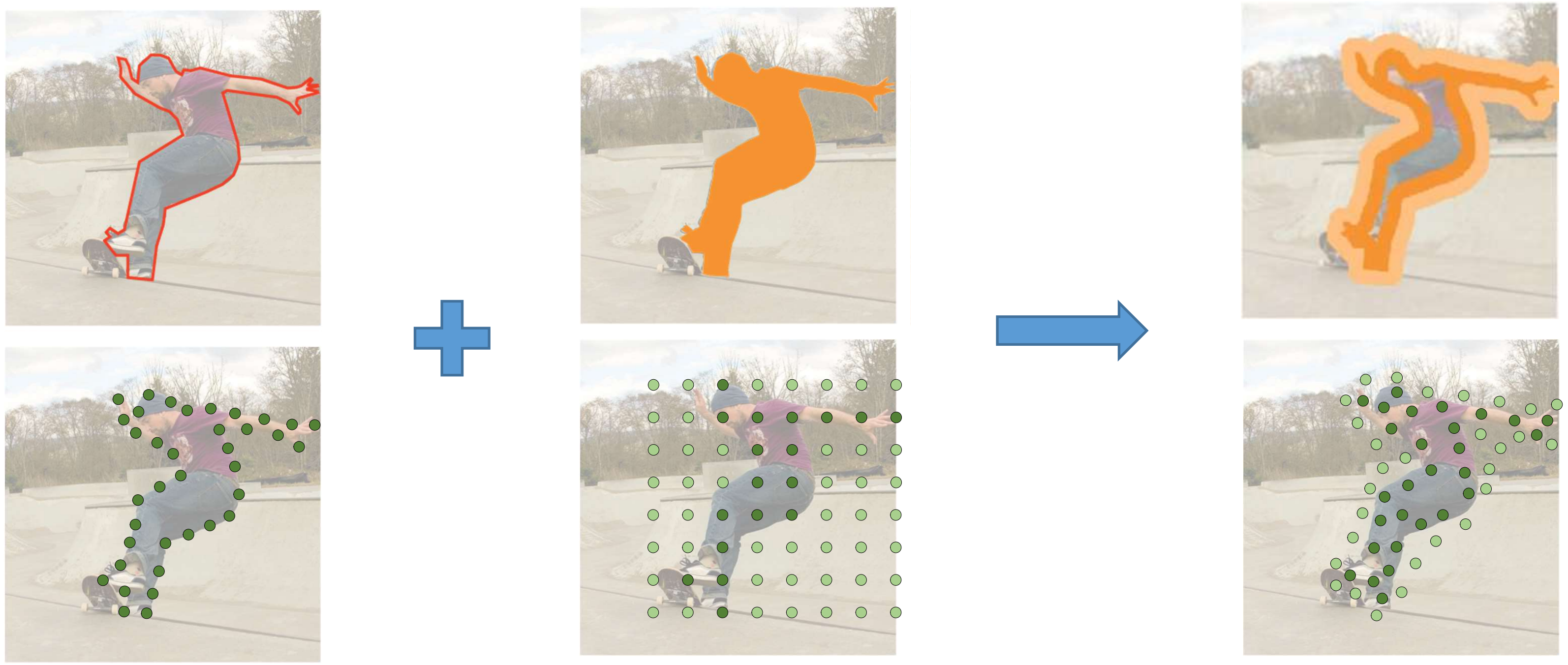
contour　boundary sample　foreground mask　grid sample

***Dense RepPoints:*** $\mathcal{R} = \left\{\left(\underline{x_i, y_i}, \boldsymbol{a_i}\right)\right\}_{i=1}^{n}$

point location　foreground score

# Dense RepPoints

A new sampling strategy, combines merits of both contour and grid mask.



contour
(boundary sample)

grid mask
(grid sample)

boundary mask
(distance transform sample)

*efficient as contour, strong as grid mask*

# Learning Dense RepPoints

Learning point set coordinates.

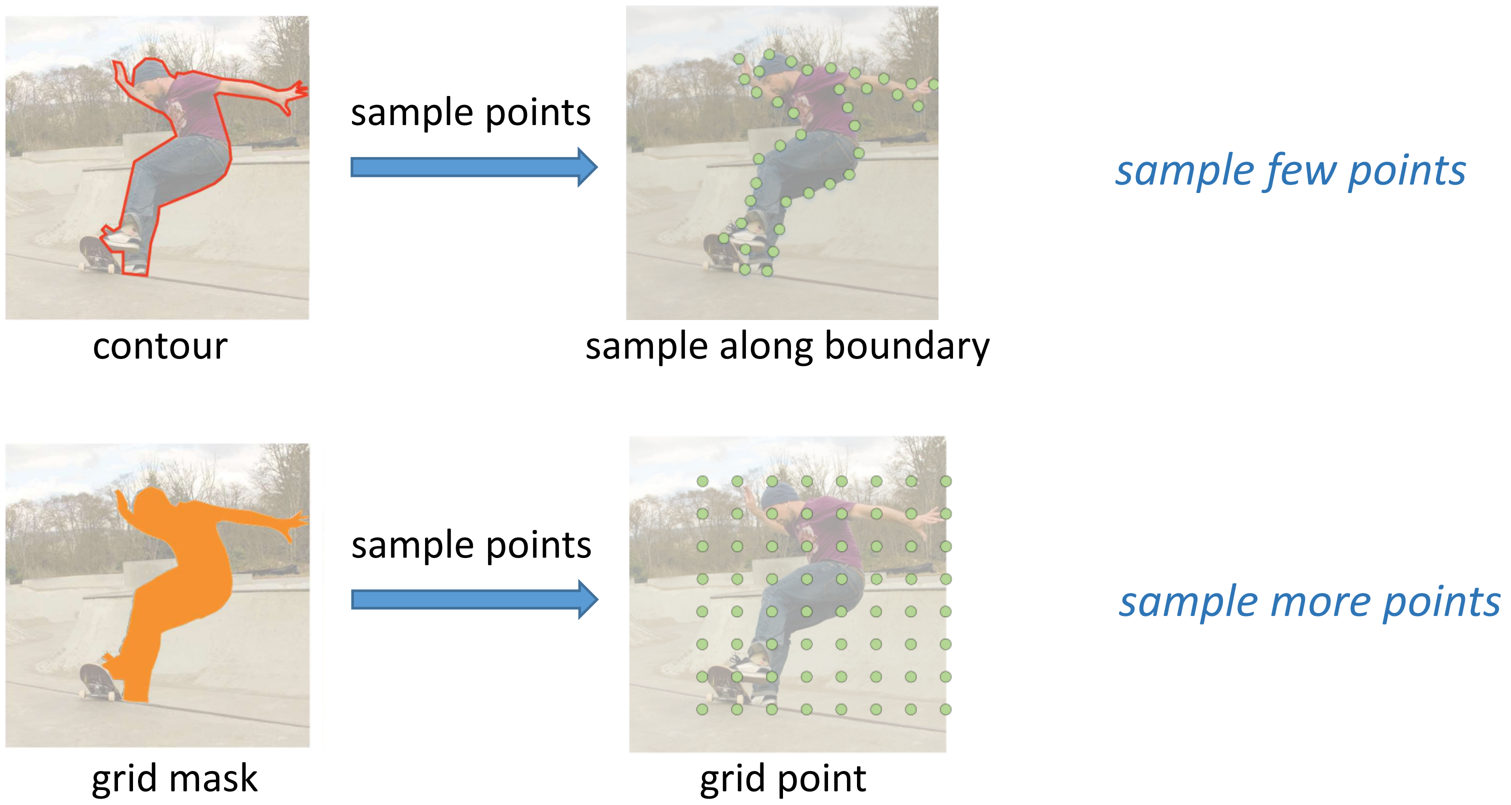
Learning per points foreground probability

Learning instance class from point set

# Learning point set coordinates.

Learning point set coordinates.

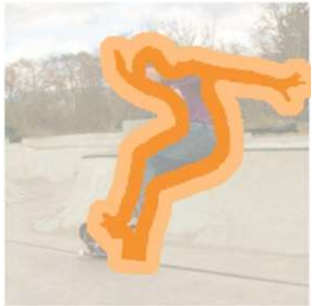
1. Sample points from GT object annotation



# Learning point set coordinates.

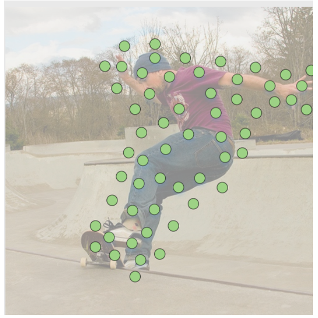
Learning point set coordinates.

1. Sample points from GT object annotation



boundary mask

sample points



distance transform sampling

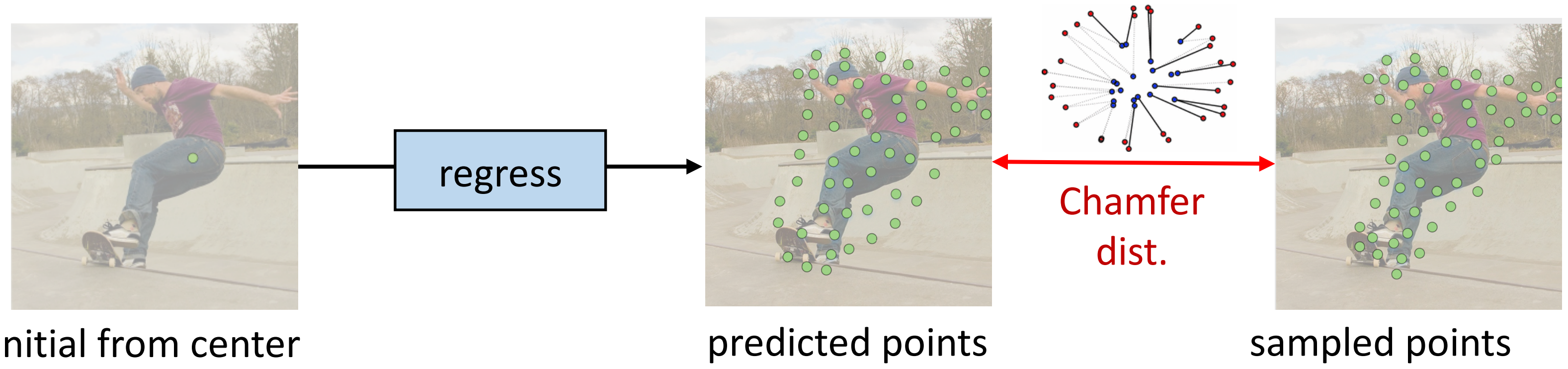
*efficient sample*

*Sample more points near object boundary*

# Learning Dense RepPoints

Learning point set coordinates.

2. Optimize the point set loss between predicted points and sampled points .




Dense RepPoints Regression:

$$\mathcal{R}_p = \{(x_i, y_i, \boldsymbol{a_i})\}_{i=1}^{n}$$

$$\mathcal{R}_{reg} = \{(x_i + \Delta x_i, y_i + \Delta y_i, \boldsymbol{a_i})\}_{i=1}^{n}$$

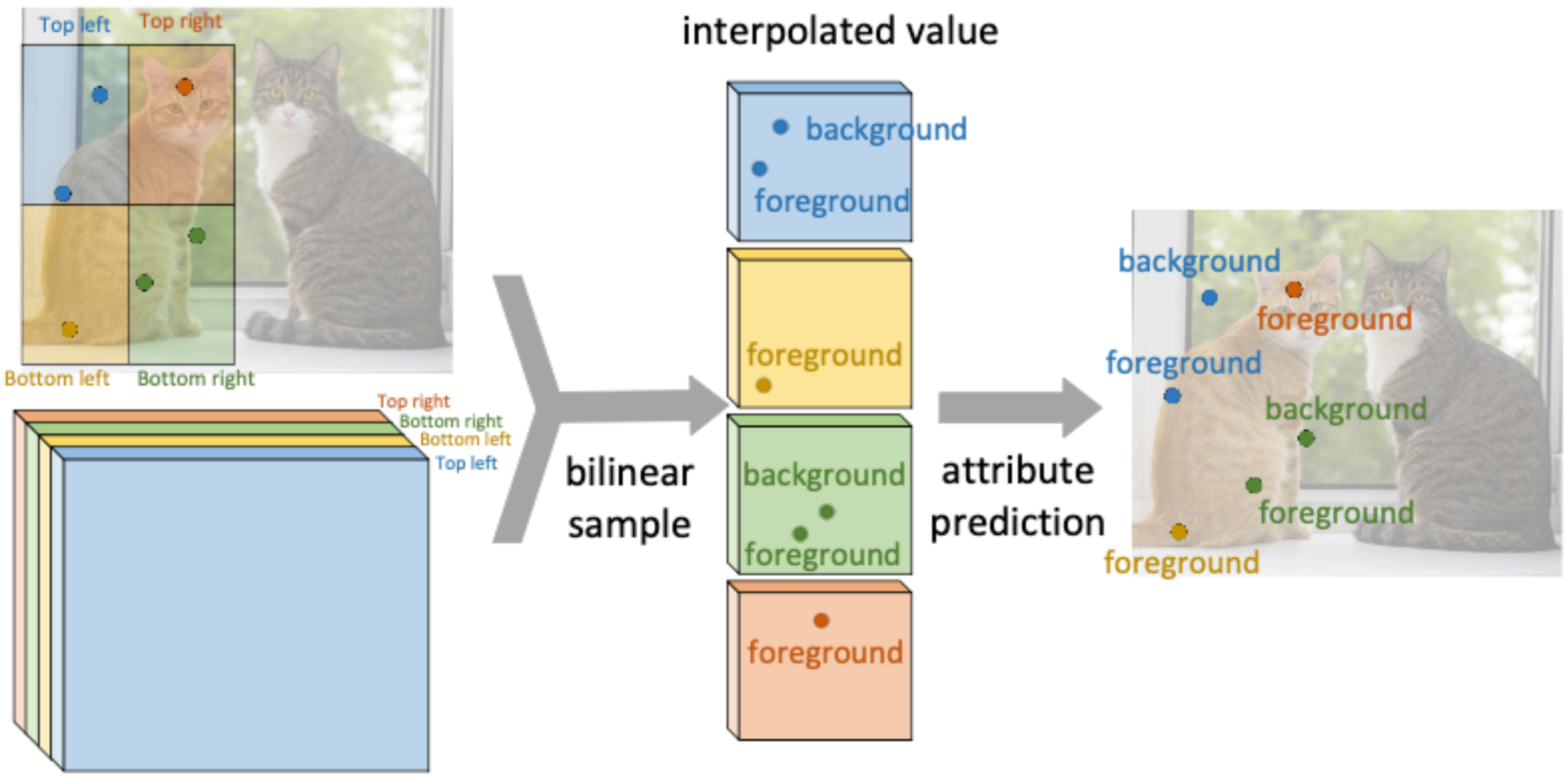
Bounding Box Regression:

$$\mathcal{B}_p = (x_p, y_p, w_p, h_p)$$

$$\mathcal{B}_{reg} = \left(x_p + w_p \Delta x_p, y_p + h_p \Delta y_p, w_p e_p^{\Delta w_p}, h_p e_p^{\Delta h_p}\right)$$

# Learning Dense RepPoints

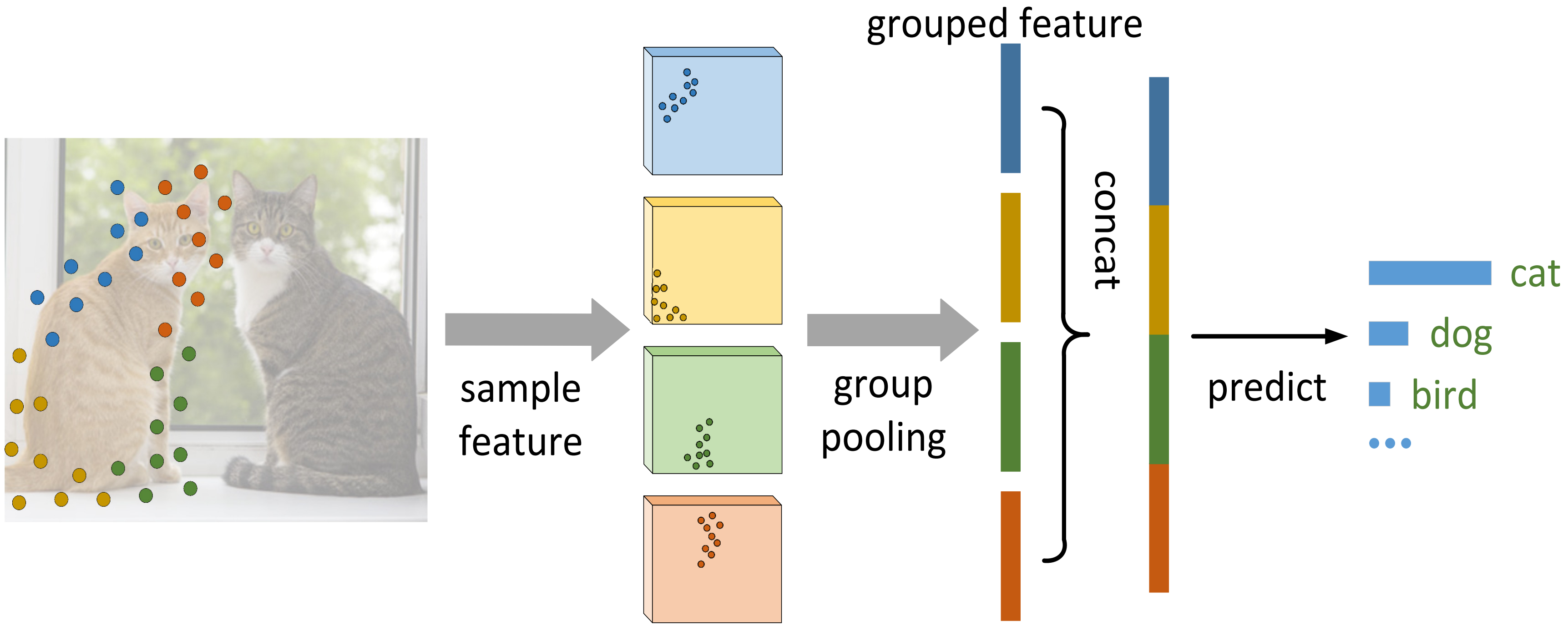
Learning per points foreground probability




*We use position-sensitive map similar to R-FCN and TensorMask.*

# Learning Dense RepPoints

Classifying the instance category from point set




*We use group pooling to reduce the computation to constant time.*

# Infer segments from Dense RepPoints

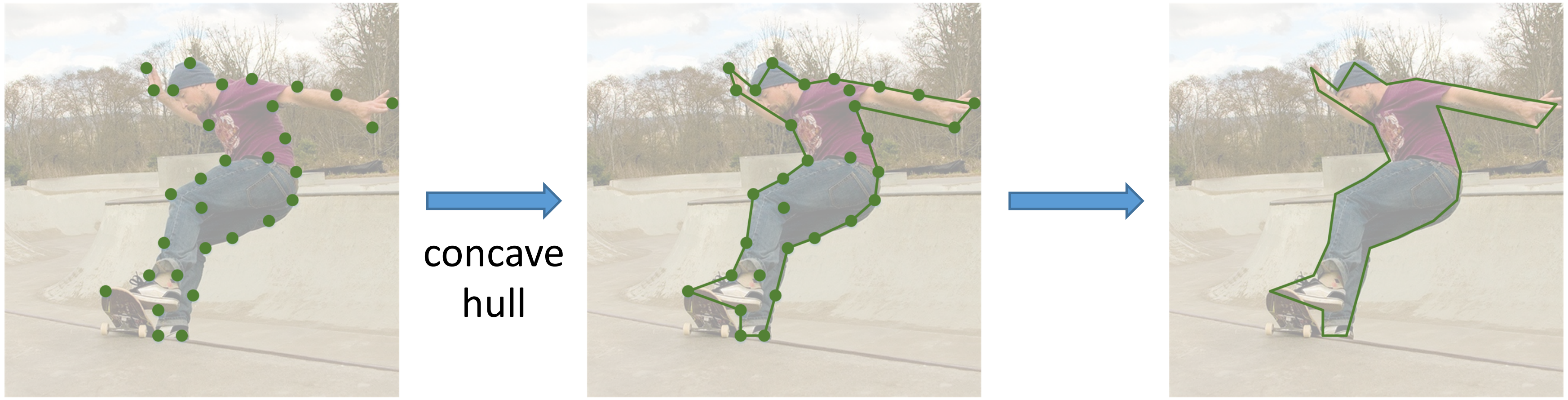
Infer from contour sampling

Infer from grid points sampling

Infer from distance transform sampling

# Inference

Inference contour using concave hull

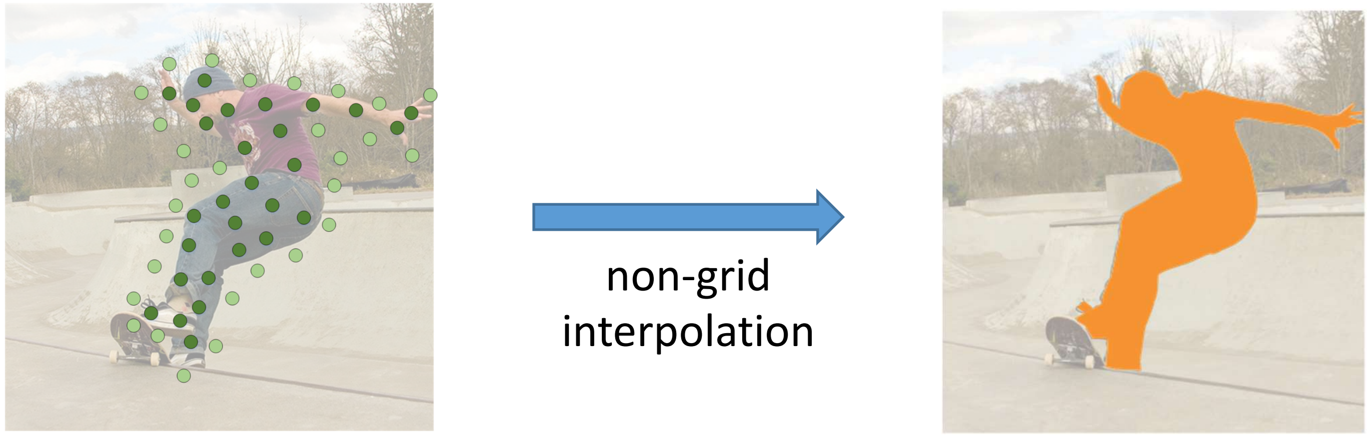


# Inference

Inference foreground mask from grid points

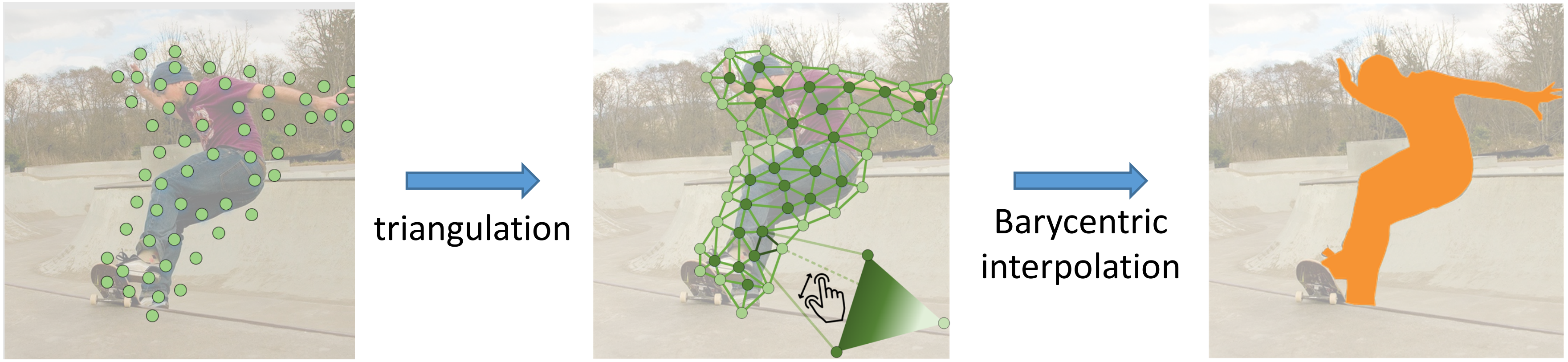

# Inference

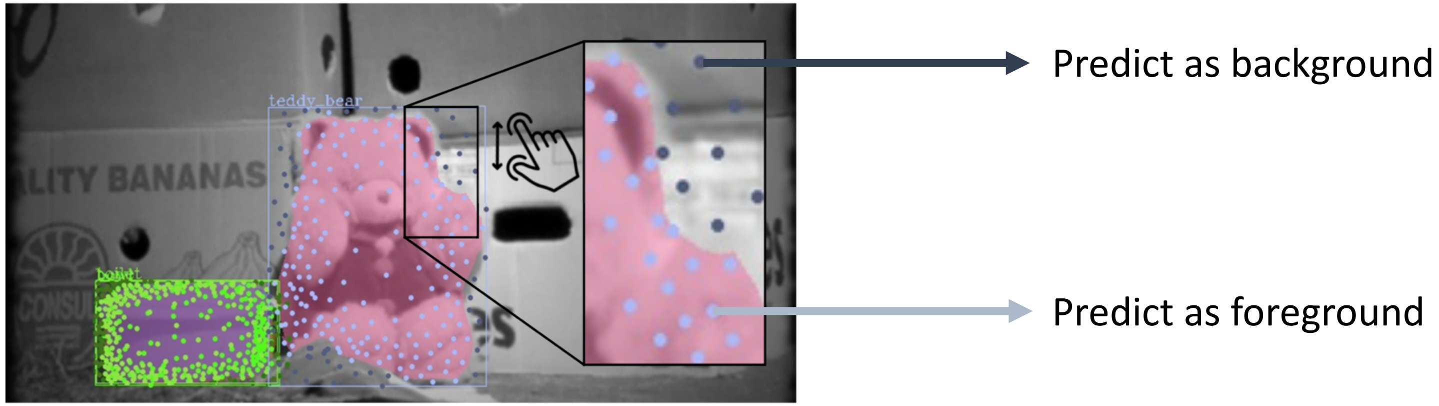
Inference foreground mask from boundary points



# Inference

Inference foreground mask from boundary points



ECCV'20 ONLINE
23-28 AUGUST 2020

# Visualization


**Top**: The learned points (225 points) is mainly distributed around the object boundary.
**Bottom**: The foreground masks generated by triangulation post-processing.

# Experiments

Ablation study

State-of-the-art comparison

# Ablation Study

Different representation of object segments

| number of points | 9 | 25 | 81 | 225 | 729 |
|---|---|---|---|---|---|
| Contour | **19.7** | 23.9 | 26.0 | 25.2 | 24.1 |
| Grid points | 5.0 | 17.6 | 29.7 | 31.6 | 32.8 |
| Boundary points | 13.9 | **24.5** | **31.5** | **32.8** | **33.8** |

*“boundary sampling” is efficient at both small and large number of points*

Number of points

| number of points | 81 | 225 | 441 | 729 |
|---|---|---|---|---|
| AP | 31.5 | 32.8 | 33.3 | **33.8** |
| AP@50 | 54.2 | 54.2 | 54.5 | **54.8** |
| AP@75 | 32.7 | 34.4 | 35.2 | **35.9** |

*Performance increase consistently with number of points, “densify” is important*

# Experiments

Instance segmentation performance

| Method | Backbone | epochs | jitter | $AP$ | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|---|---|---|
| Mask R-CNN [18] | ResNet-101 | 12 | | 35.7 | 58.0 | 37.8 | 15.5 | 38.1 | 52.4 |
| Mask R-CNN [18] | ResNeXt-101 | 12 | | 37.1 | 60.0 | 39.4 | 16.9 | 39.9 | 53.5 |
| TensorMask [7] | ResNet-101 | 72 | ✓ | 37.1 | 59.3 | 39.4 | 17.4 | 39.1 | 51.6 |
| SOLO [42] | ResNet-101 | 72 | ✓ | 37.8 | 59.5 | 40.4 | 16.4 | 40.6 | **54.2** |
| ExtremeNet [50] | HG-104 | 100 | ✓ | 18.9 | - | - | 10.4 | 20.4 | 28.3 |
| PolarMask [45] | ResNet-101 | 24 | ✓ | 32.1 | 53.7 | 33.1 | 14.7 | 33.8 | 45.3 |
| **Ours** | ResNet-101 | 36 | ✓ | **39.1** | **62.2** | **42.1** | **21.8** | **42.5** | 50.8 |

+1.3 improvement over state-of-the-art

# Experiments

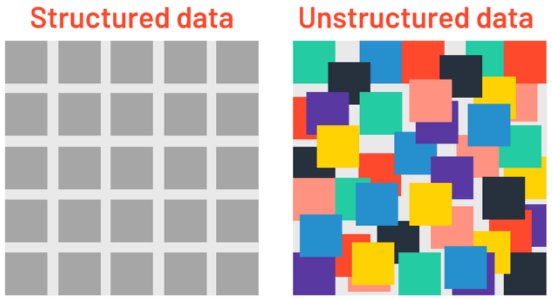
Object detection performance

| Method | Backbone | epochs | jitter | $AP$ | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|---|---|---|
| Faster R-CNN[27] | ResNet-101 | 12 | | 36.2 | 59.1 | 39.0 | 18.2 | 39.0 | 48.2 |
| Mask R-CNN[18] | ResNet-101 | 12 | | 38.2 | 60.3 | 41.7 | 20.1 | 41.1 | 50.2 |
| Mask R-CNN[18] | ResNeXt-101 | 12 | | 39.8 | 62.3 | 43.4 | 22.1 | 43.2 | 51.2 |
| RetinaNet[28] | ResNet-101 | 12 | | 39.1 | 59.1 | 42.3 | 21.8 | 42.7 | 50.2 |
| RepPoints[47] | ResNet-101 | 12 | | 41.0 | 62.9 | 44.3 | 23.6 | 44.1 | 51.7 |
| ATSS[48] | ResNeXt-101-DCN | 24 | ✓ | 47.7 | 66.5 | 51.9 | 29.7 | 50.8 | 59.4 |
| CornerNet[25] | HG-104 | 100 | ✓ | 40.5 | 56.5 | 43.1 | 19.4 | 42.7 | 53.9 |
| ExtremeNet[50] | HG-104 | 100 | ✓ | 40.1 | 55.3 | 43.2 | 20.3 | 43.2 | 53.1 |
| CenterNet [49] | HG-104 | 100 | ✓ | 42.1 | 61.1 | 45.9 | 24.1 | 45.5 | 52.8 |
| **Ours** | ResNeXt-101+DCN | 36 | ✓ | **48.9** | **69.2** | **53.4** | **30.5** | **51.9** | **61.2** |

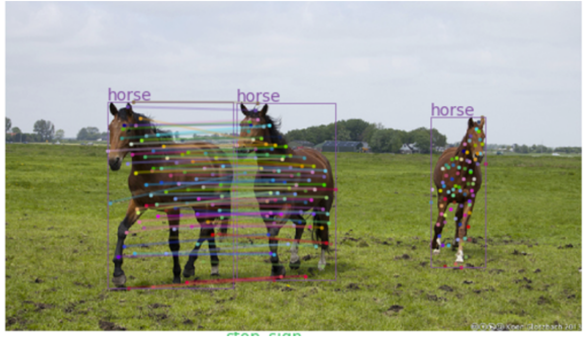
+1.2 improvement over state-of-the-art

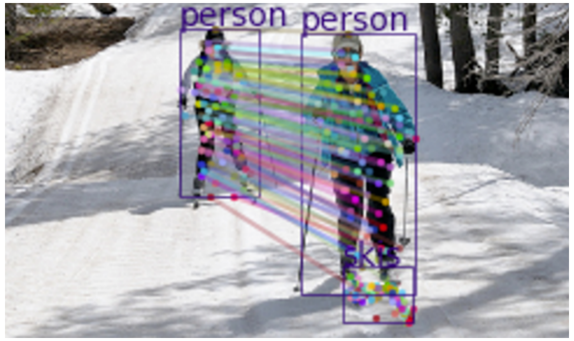
# Insights

- Unstructure data representation for 2D visual tasks, especially for high-definition media.




- Unsupervised keypoints/correspondence learning from video, simulation.




- Box-free visual perception task, e.g. key-point estimation, video tracking, etc.