Instance Segmentation

Riley Simmons-Edler, Berthy Feng
Instance Segmentation Task

- Label each foreground pixel with object and instance
- Object detection + semantic segmentation
In This Lecture...

- Microsoft COCO dataset
- Mask R-CNN (fully supervised)
- Mask$^X$ R-CNN (partially supervised)
Microsoft COCO: Common Objects in Context

Previous Datasets

- **ImageNet**: many object categories
- **PASCAL VOC**: object detection in natural images, small number of classes
- **SUN**: labeling scene types and commonly occurring objects, but not many instances per category

Image Credit: Tsung-Yi Lin et al.
Goal: Push research in scene understanding

1. Detecting non-iconic views
2. Contextual reasoning between objects
3. Precise 2D localization of objects
MS COCO Dataset

- 91 object classes
- 328,000 images
- 2.5 million labeled instances

Image Credit: Tsung-Yi Lin et al.
Image Collection & Annotation
Object Categories

Instances per category

Image Credit: Tsung-Yi Lin et al.
Non-Iconic Image Collection

(a) Iconic object images
(b) Iconic scene images
(c) Non-iconic images

Image Credit: Tsung-Yi Lin et al.
Fig. 3: Our annotation pipeline is split into 3 primary tasks: (a) labeling the categories present in the image (§4.1), (b) locating and marking all instances of the labeled categories (§4.2), and (c) segmenting each object instance (§4.3).
Dataset Evaluation
Statistics

Categories per image

Instances per image

Image Credit: Tsung-Yi Lin et al.
Statistics

Number of categories vs. number of instances

Instance size

Image Credit: Tsung-Yi Lin et al.
The COCO 2017 Detection Challenge is designed to push the state of the art in object detection forward. Teams are encouraged to compete in either (or both) of two object detection challenges: using bounding box output or object segmentation output. For full details of this task please see the COCO Detection Challenge page.
COCO Keypoint Challenge

The COCO 2017 Keypoint Challenge requires localization of person keypoints in challenging, uncontrolled conditions. The keypoint challenge involves simultaneously detecting people and localizing their keypoints (person locations are not given at test time). For full details of this task please see the COCO Keypoints Challenge page.

Image Credit: Tsung-Yi Lin et al.
COCO Stuff Challenge

The COCO 2017 Stuff Segmentation Challenge is designed to push the state of the art in semantic segmentation of stuff classes. Whereas the COCO 2017 Detection Challenge addresses thing classes (person, car, elephant), this challenge focuses on stuff classes (grass, wall, sky). For full details of this task please see the COCO Stuff Challenge page.
COCO Places Challenges

Image Credit: Tsung-Yi Lin et al.
Mask R-CNN

Faster R-CNN
Fast R-CNN

Image Credit: Shaoqing Ren et al.

Image Credit: Tomasz Grel
Insight: Region Proposal and Detection Use Same Features

Figure 2: Faster R-CNN is a single, unified network for object detection. The RPN module serves as the ‘attention’ of this unified network.

Image Credit: Shaoqing Ren et al.
Faster R-CNN = RPN + Fast R-CNN

RPN = Fully Convolutional Network
Extending to Instance Segmentation
Visual Perception Problems

Object Detection ✓
Semantic Segmentation ✓
Instance Segmentation ?
Instance Segmentation Methods

R-CNN driven

FCN driven

(proposals)
Insight: Mask Prediction in Parallel

Mask R-CNN

Faster R-CNN

Conv

RoIAlign

class box

Slide Credit: Kaiming He
RoIPool

Image Credit: Tomasz Grel
RoIPool

- RoIPool *breaks* pixel-to-pixel translation-equivariance

Slide Credit: Kaiming He
RoIAlign

conv feat. map

Grid points of bilinear interpolation

RoIAlign output

(Fixed dimensional representation)

(Variable size RoI)

Slide Credit: Kaiming He
Mask R-CNN
Mask R-CNN Results
Examples

- Mask AP = 35.7
## Comparisons

<table>
<thead>
<tr>
<th></th>
<th>backbone</th>
<th>AP</th>
<th>AP\textsubscript{50}</th>
<th>AP\textsubscript{75}</th>
<th>AP\textsubscript{S}</th>
<th>AP\textsubscript{M}</th>
<th>AP\textsubscript{L}</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNC [10]</td>
<td>ResNet-101-C4</td>
<td>24.6</td>
<td>44.3</td>
<td>24.8</td>
<td>4.7</td>
<td>25.9</td>
<td>43.6</td>
</tr>
<tr>
<td>FCIS [26] +OHEM</td>
<td>ResNet-101-C5-dilated</td>
<td>29.2</td>
<td>49.5</td>
<td>-</td>
<td>7.1</td>
<td>31.3</td>
<td>50.0</td>
</tr>
<tr>
<td>FCIS+++ [26] +OHEM</td>
<td>ResNet-101-C5-dilated</td>
<td>33.6</td>
<td>54.5</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>ResNet-101-C4</td>
<td>33.1</td>
<td>54.9</td>
<td>34.8</td>
<td>12.1</td>
<td>35.6</td>
<td>51.1</td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>ResNet-101-FPN</td>
<td>35.7</td>
<td>58.0</td>
<td>37.8</td>
<td>15.5</td>
<td>38.1</td>
<td>52.4</td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>ResNeXt-101-FPN</td>
<td>37.1</td>
<td>60.0</td>
<td>39.4</td>
<td>16.9</td>
<td>39.9</td>
<td>53.5</td>
</tr>
</tbody>
</table>

Image Credit: Kaiming He et al.
Comparisons

Figure 6. FCIS+++ [26] (top) vs. Mask R-CNN (bottom, ResNet-101-FPN). FCIS exhibits systematic artifacts on overlapping objects.

Image Credit: Kaiming He et al.
Application: Human Pose Estimation

Figure 7. Keypoint detection results on COCO test using Mask R-CNN (ResNet-50-FPN), with person segmentation masks predicted from the same model. This model has a keypoint AP of 63.1 and runs at 5 fps.

Image Credit: Kaiming He et al.
Mask R-CNN Recap

- Add parallel mask prediction head to Faster-RCNN
- RoIAlign allows for precise localization
- Mask R-CNN improves on AP of previous state-of-the-art, can be applied in human pose estimation
Learning to Segment Every Thing

Partially Supervised Model
Motivation for a Partially Supervised Model

A = set of object categories with complete mask annotations
B = set of object categories with only bounding boxes (no segmentation annotations)

How can we know $C = A \cup B$?
Transfer Learning

Image Credit: Ronghang Hu et al.
Weight Transfer Function

\[ w_{\text{seg}}^c = \mathcal{T}(w_{\text{det}}^c; \theta) \]

\[ w_{\text{det}}^c = w_{\text{box}}^c \]

\[ w_{\text{det}}^c = [w_{\text{cls}}^c, w_{\text{box}}^c] \]
Training

- Train bounding box head using standard box detection losses on all classes in $A \cup B$
- Train mask head, weight transfer function using mask loss on classes in $A$
Stage-Wise Training

1. Detection training
2. Segmentation training

- Train detection once and then fine-tune weight transfer function
- Inferior performance

Image Credit: Ronghang Hu et al.
End-to-End Joint Training

- Jointly train detection head and mask head end-to-end
- Want detection weights to stay constant between A and B

\[
w_{\text{seg}}^c = \mathcal{T}(\text{stop}_\text{grad}(w_{\text{det}}^c); \theta)
\]
End-to-End Training Better

<table>
<thead>
<tr>
<th>method</th>
<th>training</th>
<th>stop grad on $w_{det}$</th>
<th>voc $\rightarrow$ non-voc</th>
<th>non-voc $\rightarrow$ voc</th>
</tr>
</thead>
<tbody>
<tr>
<td>class-agnostic</td>
<td>sw</td>
<td>n/a</td>
<td>14.2 34.4</td>
<td>21.5 30.7</td>
</tr>
<tr>
<td>transfer</td>
<td>sw</td>
<td>n/a</td>
<td>20.2 35.2</td>
<td>26.0 31.2</td>
</tr>
<tr>
<td>class-agnostic</td>
<td>e2e</td>
<td>n/a</td>
<td>19.2 36.8</td>
<td>23.9 32.5</td>
</tr>
<tr>
<td>transfer</td>
<td>e2e</td>
<td></td>
<td>20.2 37.7</td>
<td>24.8 33.2</td>
</tr>
<tr>
<td>transfer</td>
<td>e2e</td>
<td>✓</td>
<td>22.2 37.6</td>
<td>27.6 33.1</td>
</tr>
</tbody>
</table>

(d) **Ablation on the training strategy.** We try both stage-wise (‘sw’) and end-to-end (‘e2e’) training (see §3.2), and whether to stop gradient from $\mathcal{T}$ to $w_{det}$. End-to-end training improves the results and it is crucial to stop gradient on $w_{det}$.
Mask Prediction

Baseline: Class-agonistic FCN mask prediction

Extension: FCN+MLP mask head:

<table>
<thead>
<tr>
<th>method</th>
<th>voc $\rightarrow$ non-voc</th>
<th>non-voc $\rightarrow$ voc</th>
</tr>
</thead>
<tbody>
<tr>
<td>class-agnostic</td>
<td>14.2</td>
<td>21.5</td>
</tr>
<tr>
<td>class-agnostic+MLP</td>
<td>17.1</td>
<td>22.8</td>
</tr>
<tr>
<td>transfer</td>
<td>20.2</td>
<td>26.0</td>
</tr>
<tr>
<td>transfer+MLP</td>
<td><strong>21.3</strong></td>
<td><strong>26.6</strong></td>
</tr>
</tbody>
</table>

(c) **Impact of the MLP mask branch.** Adding the class-agnostic MLP mask branch (see §3.4) improves the performance of classes in set $B$ for both the class-agnostic baseline and our weight transfer approach.
Results
Examples

Figure 4. Mask predictions from the class-agnostic baseline (top row) vs. our Mask$^X$ R-CNN approach (bottom row). Green boxes are classes in set $A$ while the red boxes are classes in set $B$. The left 2 columns are $A = \{\text{voc}\}$ and the right 2 columns are $A = \{\text{non-voc}\}$.

Image Credit: Ronghang Hu et al.
## Comparisons

| backbone  | method                  |\begin{tabular}{l|c|c|c|c|c|c} \hline voc $\rightarrow$ non-voc: test on $B = \{\text{non-voc}\}$ \hline & AP  & AP$_{50}$ & AP$_{75}$ & AP$_S$ & AP$_M$ & AP$_L$ \\ R-50-FPN & class-agnostic (baseline) & 19.2 & 36.4 & 18.4 & 11.5 & 23.3 & 24.4 \\ & Mask$^X$ R-CNN (ours) & \textbf{23.7} & \textbf{43.1} & \textbf{23.5} & \textbf{12.4} & \textbf{27.6} & \textbf{32.9} \\ & fully supervised (oracle) & 33.0 & 53.7 & 35.0 & 15.1 & 37.0 & 49.9 \\ R-101-FPN & class-agnostic (baseline) & 18.5 & 34.8 & 18.1 & 11.3 & 23.4 & 21.7 \\ & Mask$^X$ R-CNN (ours) & \textbf{23.8} & \textbf{42.9} & \textbf{23.5} & \textbf{12.7} & \textbf{28.1} & \textbf{33.5} \\ & fully supervised (oracle) & 34.4 & 55.2 & 36.3 & 15.5 & 39.0 & 52.6 \\ \hline \end{tabular} |\begin{tabular}{l|c|c|c|c|c|c} \hline non-voc $\rightarrow$ voc: test on $B = \{\text{voc}\}$ \hline & AP  & AP$_{50}$ & AP$_{75}$ & AP$_S$ & AP$_M$ & AP$_L$ \\ R-50-FPN & class-agnostic (baseline) & 23.9 & 42.9 & 23.5 & 11.6 & 24.3 & 33.7 \\ & Mask$^X$ R-CNN (ours) & \textbf{28.9} & \textbf{52.2} & \textbf{28.6} & \textbf{12.1} & \textbf{29.0} & \textbf{40.6} \\ & fully supervised (oracle) & 37.5 & 63.1 & 38.9 & 15.1 & 36.0 & 53.1 \\ R-101-FPN & class-agnostic (baseline) & 24.7 & 43.5 & 24.9 & 11.4 & 25.7 & 35.1 \\ & Mask$^X$ R-CNN (ours) & \textbf{29.5} & \textbf{52.4} & \textbf{29.7} & \textbf{13.4} & \textbf{30.2} & \textbf{41.0} \\ & fully supervised (oracle) & 39.1 & 64.5 & 41.4 & 16.3 & 38.1 & 55.1 \\ \hline \end{tabular} |

Table 2. **End-to-end training of Mask$^X$ R-CNN.** As in Table 1, we use ‘cls+box, 2-layer, LeakyReLU’ implementation of $T$ and add the MLP mask branch (‘transfer+MLP’), and follow the same evaluation protocol. We also report AP$_{50}$ and AP$_{75}$ (average precision evaluated at 0.5 and 0.75 IoU threshold respectively), and AP over small (AP$_S$), medium (AP$_M$), and large (AP$_L$) objects. Our method significantly outperforms the baseline on those classes in set $B$ without mask training data for both ResNet-50-FPN and ResNet-101-FPN backbones.
Segmenting Everything

Figure 5. Example mask predictions from our Mask R-CNN on 3000 classes in Visual Genome. The green boxes are the 80 classes that overlap with COCO (set A with mask training data) while the red boxes are the remaining 2920 classes not in COCO (set B without mask training data). It can be seen that our model generates reasonable mask predictions on many classes in set B. See §5 for details.