Pixel Consensus Voting for Panoptic Segmentation

Haochen Wang    Ruotian Luo    Michael Maire    Greg Shakhnarovich
Object detection/Instance segmentation

PASCAL VOC 2007 Bounding Box Detection

MSCOCO (2014): Bounding Box and Instance Mask
Object detection/Instance segmentation

• reason about densely enumerated bounding boxes, and then refine to instance masks.

The Mask R-CNN framework for instance segmentation
Semantic segmentation

• Label each pixel in the image with a category label
• Don’t differentiate instances, only care about pixels
Semantic segmentation
Panoptic Segmentation

- Instance-aware semantic segmentation.
- Things (*bicycle, dog, car, person*)
- Stuff (*pavement, ground, dirt, wall*)
Existing methods for panoptic segmentation

• Merge instance segmentation and semantic segmentation
However?

- Panoptic Segmentation removes the concept of boxes and focuses on pixels.
- Bounding boxes may not be the optimal intermediate representations to predict.
- Pixel Consensus Voting: pixels vote for object centroid
  - Training: per pixel classification
  - Inference: Generalized Hough Transform similar to Implicit Shape Model (ISM)

Pipeline

• Discretize regions around each pixel. CNN classifies the likely regions that contain centroid.

• Vote aggregation: probabilities at each pixel are cast to corresponding regions through dilated deconvolution.

• Peaks in voting heatmap are detections.

• Back-projection by convolving the query filter within a peak region to get an instance mask.

• Category information provided by the parallel semantic segmentation head.
Classification or Regression

- A 2d offset vector is a limited representation.
- Regression fails to capture uncertainty. Spurious peaks and false positives.
- Insight echoed by bounding box detectors: classify a proposal into anchors rather than direct regression.
Classification or Regression

• Perfect centroid prediction is unnecessary. What matters is consensus.

• Tolerance for coarse prediction depends on the scale.

• Easier to learn and train.

Pixels of a large object need only a rough estimate

Pixels of a small object need to be precise
Discretization Scheme

• Discretize regions around each pixel into radially expanding cells.

• **Voting mask** records the ground truth label if the centroid falls into a cell.

• Full discretization has 233 cells covering $243^2$ regions at 1/4 input resolution.

```
12 12 12
12 12 12
13 13 13
13 13 13
13 13 13
14 14 14
14 14 14
14 14 14
```

A toy discretization of the 9 x 9 region around location (4, 4)
There are 17 cells.

```
12 12 12
12 12 12
13 13 13
13 13 13
13 13 13
14 14 14
14 14 14
14 14 14
```

To assign voting label, align mask center with the current pixel, read off the label from the region that contains the centroid.

```
12 12 12
12 12 12
13 13 13
13 13 13
13 13 13
14 14 14
14 14 14
14 14 14
```

Full discretization of the 243^2 region around each pixel
There are 233 cells.
Voting as Dilated Deconvolution

- Spreads probabilistic votes from a point to spatial locations
- Dilation allows a pixel to send its votes afar
- Fixed kernel parameters to enable voting

Voting kernel shape [9, 1, 3, 3], dilation 3
Voting as Dilated Deconvolution

- The previous step sends votes to points, but we are voting for regions.
- Average pooling spreads the votes evenly within each cell.
- Overall voting takes 1.3ms/image over COCO val.
Peak detection

- Thresholding + connected component
Back-projection as convolutional filtering

• For a peak, find the pixels that favor this peak above all others

• Query filter: “inward reflection” of the voting mask. Convolve in a peak region to pick up instance mask.

• 81.8ms/image.

Voting mask: gt votes by a single pixel for possible centroids

Query filter: gt votes by surrounding pixels for a particular centroid
Network

Image → FPN → Semantic branch → Semantic logits → Voting logits

Trainable

Semantic branch

Voting branch
## Qualitative results

<table>
<thead>
<tr>
<th>Methods</th>
<th>Split</th>
<th>PQ</th>
<th>SQ</th>
<th>RQ</th>
<th>PQ$_{th}$</th>
<th>SQ$_{th}$</th>
<th>RQ$_{th}$</th>
<th>PQ$_{st}$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mask R-CNN</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PFPN [24] (1x)</td>
<td>val</td>
<td>39.39</td>
<td>77.83</td>
<td>48.31</td>
<td>45.91</td>
<td>80.85</td>
<td>55.35</td>
<td>29.55</td>
</tr>
<tr>
<td>PFPN [24] (3x)</td>
<td>val</td>
<td>41.48</td>
<td>79.08</td>
<td>50.52</td>
<td>48.26</td>
<td>82.21</td>
<td>57.85</td>
<td>31.24</td>
</tr>
<tr>
<td>UPSNet [62] (1x)</td>
<td>val</td>
<td>42.5</td>
<td>78.0</td>
<td>52.5</td>
<td>48.6</td>
<td>79.4</td>
<td>59.6</td>
<td>33.4</td>
</tr>
<tr>
<td><strong>Single Stage Detection</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSPS [59]</td>
<td>val</td>
<td>32.4</td>
<td>-</td>
<td>-</td>
<td>34.8</td>
<td>-</td>
<td>-</td>
<td>28.6</td>
</tr>
<tr>
<td>SSPS [59]</td>
<td>test-dev</td>
<td>32.6</td>
<td>74.3</td>
<td>42.0</td>
<td>35.0</td>
<td>74.8</td>
<td>44.8</td>
<td>29.0</td>
</tr>
<tr>
<td><strong>Proposal-Free</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AdaptIS [56]</td>
<td>val</td>
<td>35.9</td>
<td>-</td>
<td>-</td>
<td>40.3</td>
<td>-</td>
<td>-</td>
<td>29.3</td>
</tr>
<tr>
<td>DeeperLab [63]</td>
<td>val</td>
<td>33.8</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DeeperLab [63]</td>
<td>test-dev</td>
<td>34.3</td>
<td>77.1</td>
<td>43.1</td>
<td>37.5</td>
<td>77.5</td>
<td>46.8</td>
<td>29.6</td>
</tr>
<tr>
<td>SSAP [16]</td>
<td>val</td>
<td>36.5</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>SSAP [16]</td>
<td>test-dev</td>
<td>36.9</td>
<td>80.7</td>
<td>44.8</td>
<td>40.1</td>
<td>81.6</td>
<td>48.5</td>
<td>32.0</td>
</tr>
<tr>
<td><strong>Ours (1x)</strong></td>
<td>val</td>
<td>37.51</td>
<td>77.65</td>
<td>47.18</td>
<td>40.01</td>
<td>78.39</td>
<td>50.03</td>
<td>33.74</td>
</tr>
<tr>
<td><strong>Ours (1x)</strong></td>
<td>test-dev</td>
<td>37.7</td>
<td>77.8</td>
<td>47.3</td>
<td>40.7</td>
<td>78.7</td>
<td>50.7</td>
<td>33.1</td>
</tr>
</tbody>
</table>
Qualitative results
Conclusion

• Instances emerge from pixel consensus on centroid locations
• Voting as dilated deconvolution
• Back-projection as convolutional filtering
• Training reduced to pixel labeling