Instance-aware Semantic Segmentation via Multi-task Network Cascades

Jifeng Dai, Kaiming He, Jian Sun

Presented by: Xiaotian Li
Outline

- Introduction
- Related Work
- Multi-task Network Cascades
- End-to-End Training
- Experiments
- Conclusion
Introduction: Computer Vision Tasks

- Computer Vision Tasks

Classification

Classification + Localization

Object Detection

Instance Segmentation

Stanford CS231n
Introduction: Semantic Segmentation

- Semantic Segmentation
  - Predict a category label for each pixel
  - Unaware of individual object instance

Introduction: Instance Segmentation

- Instance Segmentation
  - Assign pixels to a particular instance of a class

Related Work: Fully Convolutional Networks for Semantic Segmentation

- Fully convolutional networks

Related Work: R-CNN

- **R-CNN**
  - Regions with CNN features
  - Adopt region/mask proposal methods
  - Convolutional layers are not shared

Related Work: Fast R-CNN/SPPnet

- Fast R-CNN/SPPnet
  - Share convolutional layers between proposals for an image

Girschick, "Fast R-CNN", ICCV 2015
Related Work: Faster R-CNN

- Faster R-CNN
  - Train a Region Proposal Network (RPN) to produce region proposals directly
  - No need for external region proposals!

Related Work: SDS

• Simultaneous Detection and Segmentation
  – Similar to R-CNN
  – Mask proposal methods
  – Region refinement

Related Work: Hypercolumn

- **Hypercolumn**
  - Hypercolumn at a pixel is the vector of activations of all units that lie above that pixel
  - Refinement using hypercolumns

Related Work: CFM

- Convolutional Feature Masking
  - Convolutional features are shared, similar to Fast R-CNN
  - Feature map based

• **Drawbacks:**
  
  – All these instance segmentation methods rely on mask proposal methods
  
  – Computationally expensive
  
  – Take no advantage of deeply learned features or large-scale training data
Multi-task Network Cascades

- Decompose Instance Segmentation Task into Sub-tasks:
  1. Differentiating Instance
  2. Estimating Masks
  3. Categorizing Objects
- Feature Sharing
- Causal Cascade

Multi-task Network Cascades

- Cascade Model

Multi-task Network Cascades

• Regressing Box-level Instances
  – Region Proposal Network (RPN)
  – Predict bounding box locations and objectness scores

Multi-task Network Cascades

- Regressing Box-level Instances
  - Stage 1 loss function: $L_1 = L_1(B(\theta))$

Multi-task Network Cascades

- Regressing Mask-level Instances
  - Region-of-Interest (RoI) pooling
  - Reshape boxes to fixed size
  - Perform logistic regression to the ground truth mask

Multi-task Network Cascades

- Regressing Mask-level Instances
  - Stage 2 loss function: $L_2 = L_2(M(\theta) | B(\theta))$

Multi-task Network Cascades

- Categorizing Instances
  - Two pathways concatenated to predict object class
  - Box-based pathway: directly use RoI pooled features
  - Mask-based pathway: mask out background features

Multi-task Network Cascades

- Categorizing Instances
  - Stage 3 loss function: \( L_3 = L_3(C(\theta) \mid B(\theta), M(\theta)) \)

Multi-task Network Cascades

- Cascades with More Stages
  - 5-stage cascade
  - Add class-wise bounding boxes regressors on stage 3
  - Stages 2 and 3 are performed for the second time

• The Loss Function:
  – $L(\theta) = L_1(B(\theta)) + L_2(M(\theta) \mid B(\theta)) + L_3(C(\theta) \mid B(\theta), M(\theta))$
  – Unlike traditional multi-task learning
  – Loss terms are dependent

• Main Technical challenge:
  – Apply the chain rule to the loss function
  – Spatial transfrom of a predicted box that determines RoI pooling
End-to-End Training

- The original RoI pooling layer:
  - Max pooling features inside any valid region of interest into a feature map with a fixed spatial extent of $H \times W$
  - Not differentiable w.r.t $R$

- Solution:
  - Perform RoI pooling by a differentiable RoI warping layer followed by standard max pooling
End-to-End Training

- Differentiable RoI Warping Layers
  - Perform cropping and warping operations by bilinear interpolation
  - Differentiable property
    \[
    \frac{\partial L_2}{\partial B_i} = \frac{\partial L_2}{\partial \mathcal{F}_{i}^{RoI}} \frac{\partial G}{\partial B_i} \mathcal{F}
    \]
  - Gradients involved in L3 is trivial to solve given this RoI warping module
Experiments

• Implementation Details
  – Non-maximum suppression
    • Threshold of Intersection-over-Union (IoU) ratio is 0.7
    • Top-ranked 300 boxes
  – Hyper-parameters for training
    • Use ImageNet pretrained models to initialize
    • Image-centric training framework
  – Inference
    • Use 5-stage inference for both 3-stage and 5-stage trained structures
    • Post-process similar predictions
Experiments on PASCAL VOC 2012

- Ablation Experiments on Training Strategies
  - Compare the results of different training strategies for MNCs
  - Sharing features does not directly improve accuracy
  - End-to-end training improves the accuracy
  - Training a 5-stage structure that is consistent with the inference-time usage further improves the accuracy

<table>
<thead>
<tr>
<th>training strategies</th>
<th>ZF net</th>
<th>VGG-16 net</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(a)</td>
<td>(b)</td>
</tr>
<tr>
<td>shared features?</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>end-to-end training?</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>training 5-stage cascades?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mAP@0.5 (%)</td>
<td>51.8</td>
<td>52.2</td>
</tr>
</tbody>
</table>

Experiments on PASCAL VOC 2012

- Comparisons with State-of-the-art Methods
  - Higher mAPr than previous best results
  - About two orders of magnitude faster than previous systems

<table>
<thead>
<tr>
<th>method</th>
<th>mAPr @0.5 (%)</th>
<th>mAPr @0.7 (%)</th>
<th>time/img (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>O²P [2]</td>
<td>25.2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>SDS (AlexNet) [13]</td>
<td>49.7</td>
<td>25.3</td>
<td>48</td>
</tr>
<tr>
<td>Hypercolumn [14]</td>
<td>60.0</td>
<td>40.4</td>
<td>&gt;80</td>
</tr>
<tr>
<td>CFM [7]</td>
<td>60.7</td>
<td>39.6</td>
<td>32</td>
</tr>
<tr>
<td>MNC [ours]</td>
<td><strong>63.5</strong></td>
<td><strong>41.5</strong></td>
<td><strong>0.36</strong></td>
</tr>
</tbody>
</table>

Object Detection Evaluations

- As a by product, achieve compelling object detection results

<table>
<thead>
<tr>
<th>system</th>
<th>training data</th>
<th>mAP$^b$ (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN [10]</td>
<td>VOC 12</td>
<td>62.4</td>
</tr>
<tr>
<td>Fast R-CNN [9]</td>
<td>VOC 12</td>
<td>65.7</td>
</tr>
<tr>
<td>Fast R-CNN [9]</td>
<td>VOC 07++12</td>
<td>68.4</td>
</tr>
<tr>
<td>Faster R-CNN [26]</td>
<td>VOC 12</td>
<td>67.0</td>
</tr>
<tr>
<td>Faster R-CNN [26]</td>
<td>VOC 07++12</td>
<td>70.4</td>
</tr>
<tr>
<td>MNC [ours]</td>
<td>VOC 12</td>
<td>70.9</td>
</tr>
<tr>
<td>MNC$_{box}$ [ours]</td>
<td>VOC 12</td>
<td>73.5</td>
</tr>
<tr>
<td>MNC$_{box}$ [ours]$^\dagger$</td>
<td>VOC 07++12</td>
<td><strong>75.9</strong></td>
</tr>
</tbody>
</table>

Experiments on PASCAL VOC 2012

Experiments on MS COCO

- **MS COCO Segmentation**
  - VGG-16
  - Extremely deep 101-layer network (ResNet-101)
  - Further adopt global context modeling and multi-scale testing, and ensembling
  - Final result on the test-challenge set is 28.2%/51.5%

<table>
<thead>
<tr>
<th>network</th>
<th>mAP@[.5:.95] (%)</th>
<th>mAP@.5 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGG-16 [27]</td>
<td>19.5</td>
<td>39.7</td>
</tr>
<tr>
<td>ResNet-101 [16]</td>
<td><strong>24.6</strong></td>
<td><strong>44.3</strong></td>
</tr>
</tbody>
</table>

Experiments on MS COCO

Conclusion

• Contributions
  – Task decomposition
  – Multi-task Network Cascades (MNCs)
  – Solely based on CNNs, without external modules
  – End-to-End Training
  – Fast and accurate
• Investigate in the future
  – Idea of exploiting network cascades in a multi-task learning framework maybe useful for other recognition tasks
  – Combine other successful strategies
Thank you!

Questions?
References