Relation Networks for Object Detection

Han Hu\textsuperscript{1}\textsuperscript{*}, Jiayuan Gu\textsuperscript{2}\textsuperscript{*}, Zheng Zhang\textsuperscript{1}\textsuperscript{*}, Jifeng Dai\textsuperscript{1}, and Yichen Wei\textsuperscript{1}

\textsuperscript{1}Microsoft Research Asia (MSRA)
\textsuperscript{2}Peking University

(*Equal contribution)
Relation Modeling in ConvNets

*pixel-pixel relation*

✓ *convolution*
Relation Modeling in ConvNets

- *pixel-pixel relation*
  - ![Diagram showing pixel-pixel relation with convolution](image)

- *part-part relation*
  - ![Diagram showing part-part relation with RoIPool+FC](image)

✓ convolution

✓ RoIPool+FC
Relation Modeling in ConvNets

- **pixel-pixel relation**
- **part-part relation**

- Effective and Easy to use
  - Parallel
  - Learnable
  - Require no relation supervision
  - Translational invariant
  - Stackable (convolution)

- ✓ convolution
- ✓ RoIPool+FC
Relation Modeling in ConvNets

**pixel-pixel relation**

**part-part relation**

**object-object relation**

- ✓ convolution
- ✓ RoIPool+FC
- ?
Well Recognized Problem

It is much easier to detect the *glove* if we know there is a *person*.
Rarely Studied in Deep Learning Era

Irregularities of objects
- At arbitrary image locations
- Of different scales
- Within different categories
- Of varying number across different images
Rarely Studied in Deep Learning Era

**Irregularities of objects**

- At arbitrary image locations
- Of different scales
- Within different categories
- Of varying number across different images
Goal: design a simple module to model object-object relation

Effective and Easy to use
✓ Parallel
✓ Learnable
✓ Require no relation supervision
✓ Translational invariant
✓ In-place, stackable
Object Relation Module

- **Extension** of attention module

**word-word** relation
(1D sequential)

**object-object** relation
(2D irregular)

Left figure credit by A. Vaswani et al.
Relation between Two Objects

appearance weight

\[ \text{dot product} \]

\[ (W_q) \]

\[ (W_k) \]

in standard \textit{attention} module
Relation between Two Objects

- A novel geometric weight

\[ W_q \] projection \( (W_q) \)
\[ W_k \] projection \( (W_k) \)

Dot product

Appearance weight + geometric weight

In standard **attention** module

In **object relation** module
 Relation between Two Objects

- A novel geometric weight

appearance weight

dot product

projection \((W_q)\)

projection \((W_k)\)

\[ \Delta b \]

4d bounding box regression vector

in standard **attention** module

app. + geometric weight

in **object relation** module
Relation between Two Objects

- A novel geometric weight

\[ (W_q \cdot \text{dot product} \cdot W_k) \]

appearance weight

\[ \Delta b \]

small network

\[ \max\{0, W_G \cdot \varepsilon_G(\Delta b)\} \]

4d bounding box regression vector

in standard attention module

in object relation module
Relation Aggregation

\[ f_{\text{out}}(n) = m \omega(m, n) \cdot f_{\text{in}}(m) \]

appearance + geometric weight
Multi-Branch Relation
Multi-Branch Relation

- branch #1 (person->glove)
- branch #2 (playground->glove)
- branch #N (duplicate proposals)

Output: concat

Input

 relation
 relation
 relation

branch #1
branch #2
branch #N
Object Relation Module

- residual structure

\[ \text{output} = \text{input} + \text{relation} \]

Object Relation Module

Effective and Easy to use
✓ Parallel
✓ Learnable
✓ Require no relation supervision
✓ Translational invariant
✓ Stackable
Application: Object Detection

What?

Where?

Person: 0.98

Tennis racket: 0.91
Region-based Object Detection

• Fast/Faster R-CNN

S. Ren et al. Faster R-CNN. NIPS, 2015
Region-based Object Detection

- Fast/Faster R-CNN

S. Ren et al. Faster R-CNN. NIPS, 2015
Region-based Object Detection

- Fast/Faster R-CNN

S. Ren et al. Faster R-CNN. NIPS, 2015
Region-based Object Detection

- Fast/Faster R-CNN

S. Ren et al. Faster R-CNN. NIPS, 2015
Our method: inserting object relation modules (ORMs)
Our method: inserting object relation modules (ORMs)
Learnable Duplicate Removal

Learnable ORM

Original score

Final score

Non-duplicate score
The **First** Fully End-to-End Object Detector

[Diagram showing the architecture of a fully end-to-end object detector with layers including CONVs, FC, ORM, and backpropagation steps.]
Results
Instance Recognition Experiments on COCO

<table>
<thead>
<tr>
<th>faster R-CNN</th>
<th>+ object relation modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>w.o. geometric weight</td>
<td>w.o. multi-branch</td>
</tr>
<tr>
<td>29.6</td>
<td>30.3</td>
</tr>
</tbody>
</table>

*Faster R-CNN with ResNet-50 model are used

• **+2.3 mAP** by inserting 2 ORM
• **with +3% FLOPs**
Instance Recognition Experiments on COCO

Faster R-CNN with ResNet-50 model are used

<table>
<thead>
<tr>
<th>faster R-CNN</th>
<th>+ object relation modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>w.o. geometric weight</td>
<td>w.o. multi-branch</td>
</tr>
<tr>
<td>29.6</td>
<td>30.3</td>
</tr>
</tbody>
</table>

*Faster R-CNN with ResNet-50 model are used

- More modules: 8 ORMs
Instance Recognition Experiments on COCO

**Table:**

<table>
<thead>
<tr>
<th>faster R-CNN</th>
<th>+ object relation modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>w.o. geometric weight</td>
<td>w.o. multi-branch</td>
</tr>
<tr>
<td>29.6</td>
<td>30.3</td>
</tr>
</tbody>
</table>

*Faster R-CNN with ResNet-50 model are used*

- Importance of **relative geometric weight**
Instance Recognition Experiments on COCO

- Importance of multi-branch relation

<table>
<thead>
<tr>
<th>faster R-CNN</th>
<th>+ object relation modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>w.o. geometric weight</td>
<td>w.o. multi-branch</td>
</tr>
<tr>
<td>29.6</td>
<td>30.3</td>
</tr>
</tbody>
</table>

*Faster R-CNN with ResNet-50 model are used
Instance Recognition Experiments on COCO

<table>
<thead>
<tr>
<th>faster R-CNN</th>
<th>+ object relation modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>w.o. geometric weight</td>
<td>w.o. multi-branch</td>
</tr>
<tr>
<td>29.6</td>
<td>30.3</td>
</tr>
</tbody>
</table>

*Faster R-CNN with ResNet-50 model are used

- Importance of residual connection
Duplicate Removal Experiments
Duplicate Removal Experiments

<table>
<thead>
<tr>
<th>method</th>
<th>parameters</th>
<th>mAP</th>
<th>mAP_{50}</th>
<th>mAP_{75}</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMS</td>
<td>$N_t = 0.3$</td>
<td>29.0</td>
<td>51.4</td>
<td>29.4</td>
</tr>
<tr>
<td>NMS</td>
<td>$N_t = 0.4$</td>
<td>29.4</td>
<td><strong>52.1</strong></td>
<td>29.5</td>
</tr>
<tr>
<td>NMS</td>
<td>$N_t = 0.5$</td>
<td>29.6</td>
<td>51.9</td>
<td>29.7</td>
</tr>
<tr>
<td><strong>NMS</strong></td>
<td>$N_t = 0.6$</td>
<td><strong>29.6</strong></td>
<td>50.9</td>
<td>30.1</td>
</tr>
<tr>
<td>NMS</td>
<td>$N_t = 0.7$</td>
<td>28.4</td>
<td>46.6</td>
<td><strong>30.7</strong></td>
</tr>
<tr>
<td>SoftNMS</td>
<td>$\sigma = 0.2$</td>
<td>30.0</td>
<td><strong>52.3</strong></td>
<td>30.5</td>
</tr>
<tr>
<td>SoftNMS</td>
<td>$\sigma = 0.4$</td>
<td>30.2</td>
<td>51.7</td>
<td>31.3</td>
</tr>
<tr>
<td><strong>SoftNMS</strong></td>
<td>$\sigma = 0.6$</td>
<td><strong>30.2</strong></td>
<td>50.9</td>
<td>31.6</td>
</tr>
<tr>
<td>SoftNMS</td>
<td>$\sigma = 0.8$</td>
<td>29.9</td>
<td>49.9</td>
<td><strong>31.6</strong></td>
</tr>
<tr>
<td>SoftNMS</td>
<td>$\sigma = 1.0$</td>
<td>29.7</td>
<td>49.7</td>
<td>31.6</td>
</tr>
<tr>
<td>ours</td>
<td>$\eta = 0.5$</td>
<td>30.3</td>
<td><strong>51.9</strong></td>
<td>31.5</td>
</tr>
<tr>
<td>ours</td>
<td>$\eta = 0.75$</td>
<td>30.1</td>
<td>49.0</td>
<td><strong>32.7</strong></td>
</tr>
<tr>
<td><strong>ours</strong></td>
<td>$\eta \in [0.5, 0.9]$</td>
<td><strong>30.5</strong></td>
<td>50.2</td>
<td>32.4</td>
</tr>
</tbody>
</table>

- **Noticeably** better than NMS
- **Slightly** better than SoftNMS
  [N. Bodla et al, 2017]
Fully End-to-End Object Detection

- Benefit from fully end-to-end training

<table>
<thead>
<tr>
<th>method</th>
<th>parameters</th>
<th>mAP</th>
<th>mAP_{50}</th>
<th>mAP_{75}</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMS</td>
<td>$N_t = 0.6$</td>
<td>29.6</td>
<td>50.9</td>
<td>30.1</td>
</tr>
<tr>
<td>SoftNMS</td>
<td>$\sigma = 0.6$</td>
<td>30.2</td>
<td>50.9</td>
<td>31.6</td>
</tr>
<tr>
<td>ours</td>
<td>$\eta = 0.5$</td>
<td>30.3</td>
<td>51.9</td>
<td>31.5</td>
</tr>
<tr>
<td>ours</td>
<td>$\eta = 0.75$</td>
<td>30.1</td>
<td>49.0</td>
<td>32.7</td>
</tr>
<tr>
<td>ours</td>
<td>$\eta \in [0.5, 0.9]$</td>
<td>30.5</td>
<td>50.2</td>
<td>32.4</td>
</tr>
<tr>
<td>ours (e2e)</td>
<td>$\eta \in [0.5, 0.9]$</td>
<td><strong>31.0</strong></td>
<td>51.4</td>
<td>32.8</td>
</tr>
</tbody>
</table>
Using Stronger Backbones

<table>
<thead>
<tr>
<th>backbone</th>
<th>setting</th>
<th>mAP</th>
<th>mAP(_{50})</th>
<th>mAP(_{75})</th>
<th>#. params</th>
<th>FLOPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>faster RCNN</td>
<td>2fc+SoftNMS</td>
<td>32.2/32.7</td>
<td>52.9/53.6</td>
<td>34.2/34.7</td>
<td>58.3M</td>
<td>122.2B</td>
</tr>
<tr>
<td></td>
<td>2fc+RM+SoftNMS</td>
<td>34.7/35.2</td>
<td>55.3/56.2</td>
<td>37.2/37.8</td>
<td>64.3M</td>
<td>124.6B</td>
</tr>
<tr>
<td></td>
<td>2fc+RM+e2e</td>
<td>35.2/35.4</td>
<td>55.8/56.1</td>
<td>38.2/38.5</td>
<td>64.6M</td>
<td>124.9B</td>
</tr>
<tr>
<td>FPN</td>
<td>2fc+SoftNMS</td>
<td>36.8/37.2</td>
<td>57.8/58.2</td>
<td>40.7/41.4</td>
<td>56.4M</td>
<td>145.8B</td>
</tr>
<tr>
<td></td>
<td>2fc+RM+SoftNMS</td>
<td>38.1/38.3</td>
<td>59.5/59.9</td>
<td>41.8/42.3</td>
<td>62.4M</td>
<td>157.8B</td>
</tr>
<tr>
<td></td>
<td>2fc+RM+e2e</td>
<td>38.8/38.9</td>
<td>60.3/60.5</td>
<td>42.9/43.3</td>
<td>62.8M</td>
<td>158.2B</td>
</tr>
<tr>
<td>DCN</td>
<td>2fc+SoftNMS</td>
<td>37.5/38.1</td>
<td>57.3/58.1</td>
<td>41.0/41.6</td>
<td>60.5M</td>
<td>125.0B</td>
</tr>
<tr>
<td></td>
<td>2fc+RM+SoftNMS</td>
<td>38.1/38.8</td>
<td>57.8/58.7</td>
<td>41.3/42.4</td>
<td>66.5M</td>
<td>127.4B</td>
</tr>
<tr>
<td></td>
<td>2fc+RM+e2e</td>
<td>38.5/39.0</td>
<td>57.8/58.6</td>
<td>42.0/42.9</td>
<td>66.8M</td>
<td>127.7B</td>
</tr>
</tbody>
</table>

*Faster R-CNN with ResNet-101 model are used (evaluation on minival/test-dev are reported)

- less than 10% computation overhead on all backbones
What is Learnt?
Object Pairs with High Relation Weights

**instance recognition**

**duplicate removal**

- [Image of instance recognition examples]
- [Image of duplicate removal examples]

- [Image of reference objects]
- [Image of other objects contributing high weights]
Class Co-Occurrence Information is Learnt

Class Co-occurrence Probability

Learnt Attentional Weights

\[ r = 0.90 \]
Conclusion

• A novel object relation module to model object-object relation
  ✓ Parallel
  ✓ Learnable
  ✓ Require no relation supervision
  ✓ Translational invariant
  ✓ Stackable

• Application: Object Detection
  ✓ Improves object detection accuracy
  ✓ The first fully end-to-end object detector

code:
https://github.com/msracver/Relation-Networks-for-Object-Detection