Position-Aware Recalibration Module:
Learning From Feature Semantics and Feature Position

Xu Ma, Song Fu
CAV Group, University of North Texas
Talk Pipeline

- Introduction
- Position-Aware Recalibration Module
  - Framework
  - Similarity Function
  - Semantic Normalization
  - Recalibration
  - Multi-head PRM
- Experiments
- Conclusion
Introduction

How folks understand a given scene **effectively**?

By analyzing the target representation and considering the surrounding context.
Introduction

In Computer Vision, we have the **Attention Mechanism**

Introduction

A classical Attention Mechanism: Non-Local operation

\[ y_i = \frac{1}{C(x)} \sum_{j} f(x_i, x_j) g(x_j) \]

Introduction

Such an operation is **problematic**:

- High computational complexity
- Missing position information
Problem 1: High computational complexity:

**Complexity:**
4 Conv(1024, 512), 2 matrix multiplication.

**Improvement:**
Change from query-specific to query-independent.

---

Problem 2: Missing position information:

Problem:
Even if we disrupt the spatial position of the features, there will be no change in the results for query-specific or query-independent operations.

Related Work:
1) Transformer[4] (NLP)  
   Positional Embedding

2) LRNet[5]  
   Geometry Prior

3) AANet[6]  
   Relative positional encodings

4) Explore Self-Attention[7]  
   Position encoding

---

We mainly improve CNNs by dealing with the following problems:
1) Efficiency;  2) Positional Information.

\[
y = \text{sigmoid}(\mathcal{N}(S)) \otimes x
\]
\[
s.t. \quad S = \alpha \phi(x, q) * D + \beta \phi(x, z),
\]
\[
D = f_p(|p_x - p_q|),
\]

Fig. A basic Position-Aware Recalibration Module
Similarity Function

We explore the similarity from two aspects:
1) The most distinct feature; 2) the global context.

\[ \phi (x, q) \quad \phi (x, z) \]

- **Cosine similarity:**
  \[ \phi (x_i, q) = \frac{x_i^T q}{\max (\|x_i\|_2 \cdot \|q\|_2, \epsilon)} \]

- **L1-norm similarity:**
  \[ \phi (x_i, q) = \sum_{c=1}^{C'} - |x_i^c - q^c| \]

- **Dot-product similarity:**
  \[ \phi (x_i, q) = x_i^T q \]
  (default)

Missing position information:

Our Solution: Gaussian Distribution.
Basic idea: the closer the distance, the greater the impact.

\[ f_p(|p_x - p_q|) = \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{\theta |p_x - p_q|}{d}\right)^2} \]

**Step1:** calculate the geometric relative position (in a range of [0,1]);
**Step2:** combine the width/height using learnable parameter **Theta**;
**Step3:** Encoding with a Gaussian Distribution [9].
Semantic Normalization

1) Calculate the mean/std of the similarity $S$ over spatial dimension:

$$\mu = \frac{1}{HW} \sum_{i=1}^{HW} S_i, \quad \sigma = \left( \frac{1}{HW} \sum_{i=1}^{HW} (S_i - \mu)^2 \right)^{\frac{1}{2}}$$

2) Normalize:

$$S = f_s (S) = \frac{S - \mu}{\sigma + \epsilon},$$

3) Apply affine transformation:

$$S = \lambda S + \xi$$
Recalibration

1) Rescale using $\text{Sigmoid}$ function;

2) Recalibrate using element—wise multiplication.
Is only one max-value point sufficient?

Question:

How do we decide the multiple key-points?

We do need multiple key points.
How do we decide the multiple key-points?

Two options:

(a) Select several points simultaneously
(b) Select for each group

Reason:

1) Complexity;
2) Decision of multiple points;
3) Performance;

Fig. Multiple Key-points scheme

(Reviewer's suggestion) (Ours implementation)
Multi-head PRM

Fig. Left: multi-head attention; Right: multi-head PRM. Fig
## Experiments

### Comparison on ImageNet

<table>
<thead>
<tr>
<th>Model</th>
<th>top-1 acc.</th>
<th>top-5 acc.</th>
<th>FLOPs (G)</th>
<th>Parameters (M)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet50 [He et al., 2016]</td>
<td>75.8974</td>
<td>92.7224</td>
<td>4.122</td>
<td>25.557</td>
</tr>
<tr>
<td>SE-ResNet50 [Hu et al., 2018b]</td>
<td>77.2877</td>
<td>93.6478</td>
<td>4.130</td>
<td>28.088</td>
</tr>
<tr>
<td>GE-ResNet50 [Hu et al., 2018a]</td>
<td>76.2357</td>
<td>92.9847</td>
<td>4.127</td>
<td>25.557</td>
</tr>
<tr>
<td>CBAM-ResNet50 [Woo et al., 2018]</td>
<td>77.2840</td>
<td>93.6005</td>
<td>4.139</td>
<td>28.090</td>
</tr>
<tr>
<td>GC-ResNet50 [Cao et al., 2019]</td>
<td>74.8966</td>
<td>92.2812</td>
<td>4.130</td>
<td>28.105</td>
</tr>
<tr>
<td>SGE-ResNet50 [Li et al., 2019a]</td>
<td>77.5072</td>
<td>93.6783</td>
<td>4.127</td>
<td>25.560</td>
</tr>
<tr>
<td>PRM-ResNet50 (ours)</td>
<td><strong>77.6474</strong></td>
<td>93.6418</td>
<td>4.128</td>
<td>25.560</td>
</tr>
</tbody>
</table>

Table: Comparison results of classification accuracy (%) and complexity on ImageNet.
Apply to other CNN models

<table>
<thead>
<tr>
<th>Models</th>
<th>PRM</th>
<th>top-1</th>
<th>FLOPs</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet50</td>
<td>w/o</td>
<td>75.8974</td>
<td>4.122G</td>
<td>25.56M</td>
</tr>
<tr>
<td></td>
<td>w/</td>
<td>77.6474</td>
<td>4.128G</td>
<td>25.56M</td>
</tr>
<tr>
<td>MobileNetV2</td>
<td>w/o</td>
<td>71.0320</td>
<td>0.320G</td>
<td>3.51M</td>
</tr>
<tr>
<td></td>
<td>w/</td>
<td>72.5466</td>
<td>0.321G</td>
<td>3.51M</td>
</tr>
<tr>
<td>MnasNet</td>
<td>w/o</td>
<td>71.7195</td>
<td>0.330G</td>
<td>4.38M</td>
</tr>
<tr>
<td></td>
<td>w/</td>
<td>73.0147</td>
<td>0.331G</td>
<td>4.38M</td>
</tr>
</tbody>
</table>

Table: The performance of PRM on different CNN architectures.
Experiments

Ablation: Similarity function

<table>
<thead>
<tr>
<th>network</th>
<th>similarity function</th>
<th>top-1</th>
<th>top-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resnet50</td>
<td>Cosine</td>
<td>77.6517</td>
<td>93.6711</td>
</tr>
<tr>
<td></td>
<td>L1-norm</td>
<td>77.6012</td>
<td>93.5944</td>
</tr>
<tr>
<td></td>
<td>Dotproduct</td>
<td>77.6474</td>
<td>93.6418</td>
</tr>
<tr>
<td>MobileNetV2</td>
<td>Cosine</td>
<td>72.5374</td>
<td>90.8714</td>
</tr>
<tr>
<td></td>
<td>L1-norm</td>
<td>72.5570</td>
<td>90.7993</td>
</tr>
<tr>
<td></td>
<td>Dotproduct</td>
<td>72.5466</td>
<td>90.8960</td>
</tr>
</tbody>
</table>

Table: The performance of different similarity functions.
Experiments

Ablation: Influence of each component

<table>
<thead>
<tr>
<th>Feature dependency</th>
<th>Position encoding</th>
<th>Normalization</th>
<th>top-1</th>
<th>top-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td></td>
<td>68.9632</td>
<td>88.5902</td>
</tr>
<tr>
<td>✓</td>
<td></td>
<td>✓</td>
<td>70.1863</td>
<td>89.4491</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td></td>
<td>70.2388</td>
<td>89.4970</td>
</tr>
<tr>
<td>✓</td>
<td></td>
<td>✓</td>
<td>70.4467</td>
<td>89.5653</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>70.5616</td>
<td>89.6635</td>
</tr>
</tbody>
</table>

Table: Ablation studies on each component of PRM. We conduct these ablation experiments based on ResNet-18.
Experiments

Ablation: Influence of head number

Fig: Influence of the head number in PRM.
Experiments

Visualization of PRM

Fig: Query points visualization. **From top to bottom:** selected query points, the positional mask from all groups, and the attention map generated by Grad-CAM. **From left to right:** the corresponding results of each stage in ResNet50. Best viewed in color.
Experiments

Application on high-level visual tasks

<table>
<thead>
<tr>
<th>Detector</th>
<th>Backbone</th>
<th>AP$_{50:95}$</th>
<th>AP$_{50}$</th>
<th>AP$_{75}$</th>
<th>AP$_S$</th>
<th>AP$_M$</th>
<th>AP$_L$</th>
<th>GMac</th>
<th>Parameters(M)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RetinaNet</td>
<td>ResNet50 [He et al., 2016]</td>
<td>36.2</td>
<td>55.9</td>
<td>38.5</td>
<td>19.4</td>
<td>39.8</td>
<td>48.3</td>
<td>239.32</td>
<td>37.74</td>
</tr>
<tr>
<td>RetinaNet</td>
<td>SE-ResNet50 [Hu et al., 2018b]</td>
<td>37.4</td>
<td>57.8</td>
<td>39.8</td>
<td>20.6</td>
<td>40.8</td>
<td>50.3</td>
<td>239.43</td>
<td>40.25</td>
</tr>
<tr>
<td>RetinaNet</td>
<td>PRM-ResNet50 (ours)</td>
<td><strong>37.7</strong></td>
<td><strong>58.4</strong></td>
<td>39.7</td>
<td><strong>21.4</strong></td>
<td>40.6</td>
<td><strong>50.7</strong></td>
<td>239.32</td>
<td>37.74</td>
</tr>
<tr>
<td>Cascade R-CNN</td>
<td>ResNet50 [He et al., 2016]</td>
<td>40.6</td>
<td>58.9</td>
<td>44.2</td>
<td>22.4</td>
<td>43.7</td>
<td>54.7</td>
<td>234.71</td>
<td>69.17</td>
</tr>
<tr>
<td>Cascade R-CNN</td>
<td>GC-ResNet50 [Cao et al., 2019]</td>
<td>41.1</td>
<td>59.7</td>
<td>44.6</td>
<td>23.6</td>
<td>44.1</td>
<td>54.3</td>
<td>234.82</td>
<td>71.69</td>
</tr>
<tr>
<td>Cascade R-CNN</td>
<td>PRM-ResNet50 (ours)</td>
<td><strong>42.5</strong></td>
<td><strong>61.2</strong></td>
<td><strong>46.2</strong></td>
<td><strong>24.2</strong></td>
<td><strong>45.8</strong></td>
<td><strong>56.4</strong></td>
<td>234.71</td>
<td>69.17</td>
</tr>
</tbody>
</table>

Table 4: Detection performance (%) using different backbones on the MS-COCO validation dataset.
Conclusion

• We present a new module to recalibrate the convolutional neural network.

• We mainly deal with two issues: efficiency + positional information.

• We achieve a SOTA performance, with minimal parameters/FLOPs increase.

• The new method generalize well on other visual tasks.
Contact Me

Personal Website: https://13952522076.github.io/

CAV Group: https://www.cse.unt.edu/~qingyang/research.html

Github: https://github.com/13952522076

Zhihu: https://www.zhihu.com/people/ma-xu-41

Email: xuma@my.unt.edu
Thank you.

Questions & Suggestions?