Squeeze-and-Excitation Networks

Jie Hu\textsuperscript{1,*}, Li Shen\textsuperscript{2,*}, Gang Sun\textsuperscript{1}

\textsuperscript{1} Momenta

\textsuperscript{2} Department of Engineering Science, University of Oxford
Squeeze-and-Excitation Networks (SE Nets) formed the foundation of our winner entry on ILSVRC 2017 Classification.
A convolutional filter is expected to be an informative combination

- Fusing *channel-wise* and *spatial* information
- Within *local* receptive fields
A Simple CNN
A Simple CNN

Channel dependencies are:

- Implicit: Entangled with the spatial correlation captured by the filters
- Local: Unable to exploit contextual information outside this region
Exploiting Channel Relationships

Can the representational power of a network be enhanced by *channel relationships*?

Design a new architectural unit

- Explicitly model interdependencies between the channels of convolutional features

- Feature recalibration
  - *Selectively* emphasise informative features and inhibit less useful ones
  - Use *global* information
Squeeze-and-Excitation Blocks

Given transformation $F_{tr}$: input $X \rightarrow$ feature maps $U$

- Squeeze
- Excitation
• Aggregate feature maps through spatial dimensions using global average pooling
• Generate channel-wise statistics

U can be interpreted as a collection of local descriptors whose statistics are expressive for the whole image.
Excitation: Adaptive Recalibration

- Learn a nonlinear and non-mutually-exclusive relationship between channels
- Employ a self-gating mechanism with sigmoid function
  - Input: channel-wise statistics
  - Bottleneck configuration with two FC layers around non-linearity
  - Output: channel-wise activations
**Excitation: Adaptive Recalibration**

- Rescale the feature maps $U$ with the channel activations $q$
  - Act on the channels of $U$
  - Channel-wise multiplication

SE blocks intrinsically introduce dynamics conditioned on the input.
Example Models

**Inception Module**

- Input: $X$
- Output: $\tilde{X}$

**SE-Inception Module**

- Input: $X$
- Output: $\tilde{X}$
  - Inception
  - Global pooling
  - FC
  - ReLU
  - FC
  - Sigmoid
  - Scale

**ResNet Module**

- Input: $X$
- Output: $\tilde{X}$
  - Residual
  - Global pooling
  - FC
  - ReLU
  - FC
  - Sigmoid
  - Scale

**SE-ResNet Module**

- Input: $X$
- Output: $\tilde{X}$
  - Residual
  - Global pooling
  - FC
  - ReLU
  - FC
  - Sigmoid
  - Scale
Experiments on ImageNet-1k dataset

• Benefits at different depths
• Incorporation with modern architectures
SE blocks consistently improve performance across different depths at minimal additional computational complexity (no more than 0.26%).

- SE-ResNet-50 exceeds ResNet-50 by 0.86% and approaches the result of ResNet-101.
Incorporation with Modern Architectures

SE blocks can boost the performance of a variety of network architectures on both residual and non-residual settings.

<table>
<thead>
<tr>
<th>Model</th>
<th>top-1 error</th>
<th></th>
<th>top-5 error</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>plain</td>
<td>SENet</td>
<td>plain</td>
<td>SENet</td>
</tr>
<tr>
<td>ResNeXt-50 [47]</td>
<td>22.11</td>
<td>21.10&lt;sup&gt;(1.01)&lt;/sup&gt;</td>
<td>5.90</td>
<td>5.49&lt;sup&gt;(0.41)&lt;/sup&gt;</td>
</tr>
<tr>
<td>ResNeXt-101 [47]</td>
<td>21.18</td>
<td>20.70&lt;sup&gt;(0.48)&lt;/sup&gt;</td>
<td>5.57</td>
<td>5.01&lt;sup&gt;(0.56)&lt;/sup&gt;</td>
</tr>
<tr>
<td>VGG-16 [39]</td>
<td>27.02</td>
<td>25.22&lt;sup&gt;(1.80)&lt;/sup&gt;</td>
<td>8.81</td>
<td>7.70&lt;sup&gt;(1.11)&lt;/sup&gt;</td>
</tr>
<tr>
<td>BN-Inception [16]</td>
<td>25.38</td>
<td>24.23&lt;sup&gt;(1.15)&lt;/sup&gt;</td>
<td>7.89</td>
<td>7.14&lt;sup&gt;(0.75)&lt;/sup&gt;</td>
</tr>
<tr>
<td>Inception-ResNet-v2 [42]</td>
<td>20.37</td>
<td>19.80&lt;sup&gt;(0.57)&lt;/sup&gt;</td>
<td>5.21</td>
<td>4.79&lt;sup&gt;(0.42)&lt;/sup&gt;</td>
</tr>
<tr>
<td>MobileNet [13]</td>
<td>29.1</td>
<td>25.3&lt;sup&gt;(3.8)&lt;/sup&gt;</td>
<td>10.1</td>
<td>7.9&lt;sup&gt;(2.2)&lt;/sup&gt;</td>
</tr>
<tr>
<td>ShuffleNet [52]</td>
<td>33.9</td>
<td>31.7&lt;sup&gt;(2.2)&lt;/sup&gt;</td>
<td>13.6</td>
<td>11.7&lt;sup&gt;(1.9)&lt;/sup&gt;</td>
</tr>
</tbody>
</table>
SE blocks can generalise well on different datasets and tasks.

- **Places365-Challenge Scene Classification**

<table>
<thead>
<tr>
<th>Model</th>
<th>top-1 err.</th>
<th>top-5 err.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Places-365-CNN [37]</td>
<td>41.07</td>
<td>11.48</td>
</tr>
<tr>
<td>ResNet-152 (ours)</td>
<td>41.15</td>
<td>11.61</td>
</tr>
<tr>
<td>SE-ResNet-152</td>
<td><strong>40.37</strong></td>
<td><strong>11.01</strong></td>
</tr>
</tbody>
</table>

  Single-crop error rates (%) on Places365 validation set.

- **Object Detection on COCO**

<table>
<thead>
<tr>
<th>Model</th>
<th>AP@IoU=0.5</th>
<th>AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet-50</td>
<td>45.2</td>
<td>25.1</td>
</tr>
<tr>
<td>SE-ResNet-50</td>
<td>46.8</td>
<td>26.4</td>
</tr>
<tr>
<td>ResNet-101</td>
<td>48.4</td>
<td>27.2</td>
</tr>
<tr>
<td>SE-ResNet-101</td>
<td>49.2</td>
<td>27.9</td>
</tr>
</tbody>
</table>

  Object detection results on the COCO 40k validation set by using the basic Faster R-CNN.
Role of Excitation

The role at different depths adapts to the needs of the network

- Early layers: Excite informative features in a class agnostic manner

![Graph SE_2_3](image)

![Graph SE_3_4](image)
The role at different depths adapts to the needs of the network

- Later layers: Respond to different inputs in a highly \textit{class-specific} manner
Conclusion

• Designed a novel architectural unit to improve the representational capacity of networks by dynamic channel-wise feature recalibration.

• Provided insights into the limitations of previous CNN architectures in modelling channel dependencies.

• Induced feature importance may be helpful to related fields, e.g. network compression.

Code and Models: [https://github.com/hujie-frank/SENet](https://github.com/hujie-frank/SENet)
Thank you!