Global average pooling in deep ConvNets

Matthieu Cord
Joint work with Thibaut Durand and Nicolas Thome*

Sorbonne Universities, UPMC Paris 6, CNRS
*CEDRIC, CNAM
Outline

1. Deep net framework
2. Fully Convolutional Nets
3. Where is Pooling inside the architecture?
4. How to pool?
**Deep Convolutional Neural Networks (Deep ConvNets)**

- **Convolution** uses local weights shared across the whole image.
- **Pooling** shrinks the spatial dimensions.

---

[LeCun-89] [Fukushima 79] [Hinton-12]
Post 2012 deep architectures

VGG, 16/19 layers, 2014

GoogleNet, 22 layers, 2014

ResNet, 152 layers, 2015
Key issues for Deep&Vision

- Computer Vision: from the ImageNet Object recognition task
  - **Classification**: How to do for large and complex scenes?
  - Detection: R-CNN Fast/Faster R-CNN
    [Girshick, CVPR14, ICCV 15, NIPS 15]
  - Segmentation

- Supervised/Unsupervised – learning generic data representation
- Theoretical support to understand deep: convergence, why it works,…
- Vision and Language
- Connection to Computational/informational Neurosciences
- Compression/Embedded/Green nets
- Deep generative models,
- …
How to deal with complex scenes?

- Working on datasets with complex scenes (large and cluttered background), not centered objects, variable size, ...

VOC07/12  MIT67  15 Scene  COCO  VOC12 Action
From ImageNet to complex scenes?

- Naive approach: resize the image
- Region based approach: use regions to have images that look like ImageNet [Oquab, CVPR14]

<table>
<thead>
<tr>
<th></th>
<th>Naive</th>
<th>Region</th>
</tr>
</thead>
<tbody>
<tr>
<td>VOC 2012 (AP)</td>
<td>70.9 %</td>
<td>78.7 %</td>
</tr>
</tbody>
</table>

- Regions $\rightarrow$ better prediction

- Full annotations expensive $\rightarrow$ training with weak supervision
From ImageNet to complex scenes?

• Working on datasets with complex scenes (large and cluttered background), not centered objects, variable size, ...

VOC07/12  MIT67  15 Scene  COCO  VOC12 Action

• Select relevant regions $\rightarrow$ better prediction

• Full annotations expensive $\Rightarrow$ training with weak supervision
Outline

1. Deep net framework
2. Fully Convolutional Nets
3. Where is Pooling inside the architecture?
4. How to pool?
How to adapt VGG scheme for large images?

VGG-16

Input image: fixed size

224 × 224 × 3 → 224 × 224 × 64

112 × 112 × 128 → 56 × 56 × 256

28 × 28 × 512 → 14 × 14 × 512

7 × 7 × 512 → 1 × 1 × 4096 → 1 × 1 × 1000

- convolution + ReLU
- max pooling
- fully connected + ReLU
- softmax
Sliding window [Sermanet, OverFeat14]
Sliding window => Convolutional Layers
WELDON: MANTRA adaptation for deep CNN

Problem

• Fixed-size image as input

Adapt architecture to weakly supervised learning

1. Fully connected layers

![Diagram showing fully connected as convolutional layer (here 4096 conv. filters 7x7x512)](image)

Fully connected as convolutional layer (here 4096 conv. filters 7x7x512)
Sliding window => Convolutional Layers

$h' \times w' \times 3$

$h' \times w' \times 64$

$h' \times w' \times 128$

$h' \times w' \times 256$

$h' \times w' \times 512$

$h' \times w' \times 512$

$h' \times w' \times 512$

$h' \times w' \times 4096$

$h = \frac{h'}{32} - 6$

$w = \frac{w'}{32} - 6$

[Diagram showing convolution layers with convolution + ReLU and max pooling highlighted]
Fully connected layers as conv layers

In many archi to process large images/datasets

- OverFeat (Sermanet)
- Fast R-CNN (Girshick)
- Weldon (Durand)
- SPLeap++ (Kulkarni)
Outline

1. Deep net framework
2. Fully Convolutional Nets
3. Where is Pooling inside the architecture?
4. How to pool?
Transfer/Pooling/Classify

- Image-based strategy
- Region-based strategy
Transfer/Pooling

Global Average Pooling [Zhou, 2016], (ResNet)

➤ Image-based strategy


*Learning Deep Features for Discriminative Localization.* CVPR 2016
Transfer/Pooling

Deep MIL [Oquab, CVPR15]

WELDON [Durand, CVPR16] (≈ProNet [Sun, CVPR16])

➢ Region-based strategy
Class Activation Mapping (CAM) for GAP [Zhou, CVPR16]
CAM
for [Oquab, CVPR15]
CAM for WELDON [Durand, CVPR16]
Outline

1. Deep net framework
2. Fully Convolutional Nets
3. Where is Pooling inside the architecture?
4. How to pool?
Pooling schemes

- Max [Oquab, CVPR15]
  \[ y^c = \max_{i,j} z^c_{i,j} \]

- GAP [Zhou, CVPR16]
  \[ y^c = \frac{1}{N} \sum_{i,j} z^c_{i,j} \]

  \[ y^c = \frac{1}{\beta} \log \left( \frac{1}{N} \sum_{i,j} \exp(\beta \cdot z^c_{i,j}) \right) \]
WELDON: max+min pooling

• $h^+$: presence of the class $\rightarrow$ high $h^+$
• $h^-$: localized evidence of the absence of class
WELDON Pooling

- max + min strategy
- Top instances: using several regions, more robust region selection [Vasconcelos, CVPR15]
WELDON Pooling

- max + min strategy
- Top instances: using several regions, more robust region selection [Vasconcelos, CVPR15]

\[
y^c = s_{k^+}^{\text{top}}(z^c) + s_{k^-}^{\text{low}}(z^c)
\]

\[
s_{k^+}^{\text{top}}(z^c) = \frac{1}{k^+} \sum_{i,j} h_{ij}^c z_{ij}^c \quad \text{with} \quad h^c = \arg \max_{h \in [h_{ij} \in \{0,1\}]} \sum_{i,j} h_{ij} z_{ij}^c \quad \text{s.t.} \quad \sum_{i,j} h_{ij} = k^+
\]

\[
s_{k^-}^{\text{low}}(z^c) = \frac{1}{k^-} \sum_{i,j} \bar{h}_{ij}^c z_{ij}^c \quad \text{with} \quad \bar{h}^c = \arg \min_{h \in [h_{ij} \in \{0,1\}]} \sum_{i,j} h_{ij} z_{ij}^c \quad \text{s.t.} \quad \sum_{i,j} h_{ij} = k^-
\]
WELDON [Durand, CVPR16]
Outline

1. Deep net framework
2. Fully Convolutional Nets
3. Where is Pooling inside the architecture?
4. How to pool?
5. Visualization and Experiments
Pooling Analysis

VOC 2007
- WELDON
- max
- LSE
- MANTRA
- GAP

VOC 2012
- WELDON
- max
- LSE
- MANTRA
- GAP

VOC 2012 Action
- WELDON
- max
- LSE
- MANTRA
- GAP

MS COCO
- WELDON
- max
- LSE
- MANTRA
- GAP

MIT67
- WELDON
- max
- LSE
- MANTRA
- GAP

CUB-200
- WELDON
- max
- LSE
- MANTRA
- GAP

Feature extraction network

Classification layer

k-max+k-min pooling

$h \times w$
### ImageNet (single model)

<table>
<thead>
<tr>
<th>Model</th>
<th>Top-1 error</th>
<th>Top-5 error</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGG16 (144 crops)</td>
<td>24.4</td>
<td>7.2</td>
</tr>
<tr>
<td>GoogleNet (144 crops)</td>
<td>-</td>
<td>7.89</td>
</tr>
<tr>
<td>GoogleNet-GAP</td>
<td>35.0</td>
<td>13.2</td>
</tr>
<tr>
<td>VGG16-GAP</td>
<td>33.4</td>
<td>12.2</td>
</tr>
<tr>
<td>Inception-ResNet-v2 (12 crops)</td>
<td>18.7</td>
<td>4.1</td>
</tr>
<tr>
<td>ResNeXt-101 (1 crop)</td>
<td>19.1</td>
<td>4.4</td>
</tr>
<tr>
<td>ResNet-101 (1 crop)</td>
<td>22.44</td>
<td>6.21</td>
</tr>
<tr>
<td>ResNet-101 (10 crops)</td>
<td>21.08</td>
<td>5.35</td>
</tr>
<tr>
<td>ResNet-152 (10 crops)</td>
<td>20.69</td>
<td>5.21</td>
</tr>
<tr>
<td>ResNet-200 (10 crops)</td>
<td>20.15</td>
<td>4.93</td>
</tr>
<tr>
<td>FCN-WELDON</td>
<td>19.21</td>
<td>4.23</td>
</tr>
</tbody>
</table>
WELDON Visual results (VOC12)
Visual results (MIT67)

<table>
<thead>
<tr>
<th>True class</th>
<th>Wrong class</th>
</tr>
</thead>
<tbody>
<tr>
<td>restaurant kitchen (1.4)</td>
<td>dining room (-0.2)</td>
</tr>
<tr>
<td>bar (1.7)</td>
<td>grocery store (0.3)</td>
</tr>
</tbody>
</table>
Extension: Segmentation

• WSL segmentation framework
  – Learning with image-level labels (presence/absence of the class)
  – Difficult task: no information about location and extend of objects

• Localized features in spatial maps

• Deep + fully connected CRFs
## Extension: Segmentation

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIL-FCN [Pathak, ICLR15]</td>
<td>24.9</td>
</tr>
<tr>
<td>MIL-Base+ILP+SP-sppxl [Pinheiro, CVPR15]</td>
<td>36.6</td>
</tr>
<tr>
<td>EM-Adapt + FC-CRF [Papandreou, ICCV15]</td>
<td>33.8</td>
</tr>
<tr>
<td>CCNN + FC-CRF [Pathak, ICCV15]</td>
<td>35.3</td>
</tr>
<tr>
<td>WILDCAT + FC-CRF</td>
<td>43.7</td>
</tr>
</tbody>
</table>

![Images of original images, ground truth (GT), heatmaps, and predictions for different methods.](image-url)
Extension: Wildcat (sub. CVPR17)

Feature extraction network

Classification layer

Class-wise pooling

k-max+k-min pooling

Feature extraction network

FCN: ResNet-101

With T. Durand, T. Mordan

Share ideas of localized feature maps with R-FCN strategy of J. Dai, Yi Li, K. He, Jian Sun: R-FCN: Object Detection via Region-based Fully Convolutional Networks [NIPS 16]
Conclusion

Global Spatial Pooling: a major component in net design

Is there any learning trick behind this?

– [Lampert, ECCV16]: seed strategy better than GAP for segmentation!

– GAP: AP better than Max pooling strategy, from 1 to 400 feedback updates
Few Team’s refs. on Deep learning for Visual Recognition

- MANTRA: Minimum Maximum Latent Structural SVM for Image Classification and Ranking, T Durand, N Thome, M Cord, ICCV 2015
- LR-CNN for fine-grained classification with varying resolution, M Chevalier+, ICIP 2015
- Top-Down Regularization of Deep Belief Networks, H. Goh, N. Thome, M. Cord, JH. Lim, NIPS 2013
- Sequentially generated instance-dependent image representations for classification, G Dulac-Arnold, L Denoyer, N Thome, M Cord, P Gallinari, ICLR 2014
- Learning Deep Hierarchical Visual Feature Coding, H. Goh+, IEEE Transactions on Neural Networks and Learning Systems 2014
- Unsupervised and supervised visual codes with Restricted Boltzmann Machines, H. Goh+, ECCV 2012
- Biasing Restricted Boltzmann Machines to Manipulate Latent Selectivity and Sparsity, H. Goh+, NIPS workshop 2010
Fast(er) R-CNN

WELDON
• Multi-scale: 8 scales (combination with Object Bank strategy)
WELDON: learning

- Objective function for multi-class task and $k = 1$:

$$ \min_w \mathcal{R}(w) + \frac{1}{N} \sum_{i=1}^{N} \ell(f_w(x_i), y_{i}^{gt}) $$

$$ f_w(x_i) = \arg \max_y \left( \max_h L_w^{\text{conv}}(x_i, y, h) + \min_{h'} L_w^{\text{conv}}(x_i, y, h') \right) $$

How to learn deep architecture?

- Stochastic gradient descent training.
- Back-propagation of the selecting windows error.
WELDON: learning

Class is present

- **Increase** score of selecting windows.

*Figure: Car map*
WELDON: learning

Class is **absent**

- **Decrease** score of selecting windows.

**Figure:** Boat map
Conclusion: connections to others Latent Variables Models

- Hidden CRF (HCRF) [Quattoni, PAMI07]

\[
\frac{1}{2} \|w\|^2 + \frac{C}{N} \sum_{i=1}^{N} \log \sum_{(y, h) \in Y \times H} \exp \langle w, \psi(x_i, y, h) \rangle - \log \sum_{h \in H} \exp \langle w, \psi(x_i, y_i, h) \rangle
\]

- Latent Structural SVM (LSSVM) [Yu, ICML09]

\[
\frac{1}{2} \|w\|^2 + \frac{C}{N} \sum_{i=1}^{N} \max_{(y, h) \in Y \times H} \{ \Delta(y_i, y) + \langle w, \psi(x_i, y, h) \rangle \} - \max_{h \in H} \langle w, \psi(x_i, y_i, h) \rangle
\]

- Marginal Structural SVM (MSSVM) [Ping, ICML14]

\[
\frac{1}{2} \|w\|^2 + \frac{C}{N} \sum_{i=1}^{N} \max_y \left\{ \Delta(y_i, y) + \log \sum_{h \in H} \exp \langle w, \psi(x_i, y, h) \rangle \right\} - \log \sum_{h \in H} \exp \langle w, \psi(x_i, y_i, h) \rangle
\]

- WELDON

\[
\frac{1}{2} \|w\|^2 + \frac{C}{N} \sum_{i=1}^{N} \max_y \left\{ \Delta(y_i, y) + \sum_{h \in \Omega \subseteq H} \langle w, \psi(x_i, y, h) \rangle \right\} - \sum_{h \in \Omega \subseteq H} \langle w, \psi(x_i, y_i, h) \rangle
\]
MANTRA: model training

Learning formulation

- Loss function: \( \ell_w(x_i, y_i) = \max_{y \in Y} [\Delta(y_i, y) + D_w(x_i, y)] - D_w(x_i, y_i) \)
  - (Margin rescaling) upper bound of \( \Delta(y_i, \hat{y}) \), constraints:
    \[
    \forall y \neq y_i, \quad \underbrace{D_w(x_i, y_i)}_{\text{score for ground truth output}} \geq \underbrace{\Delta(y_i, y)}_{\text{margin}} + \underbrace{D_w(x_i, y)}_{\text{score for other output}}
    \]

- Non-convex optimization problem

\[
\min_w \frac{1}{2} \|w\|^2 + \frac{C}{N} \sum_{i=1}^{N} \ell_w(x_i, y_i) \tag{3}
\]

- Solver: non convex one slack cutting plane [Do, JMLR12]