Deep Fisher Networks and Class Saliency Maps for Object Classification and Localisation

Karáén Simonyan, Andrea Vedaldi, Andrew Zisserman
Visual Geometry Group, University of Oxford
Outline

• Classification challenge
  • can Fisher Vector encodings be improved by a deep architecture?
  • deep Fisher Network (FN)
  • combination of two deep models: Convolutional Network (CN) and deep Fisher Network

• Localisation challenge
  • visualization of class saliency maps and per-image foreground pixels from a single classification CN
  • bounding boxes computed from foreground pixels
  • weak supervision: only image class labels used for training
Shallow Image Encoding & Classification

- Dense SIFT features

- Bag of Visual Words (BOW) pipeline

[Luong & Malik, 1999]
[Varma & Zisserman, 2003]
[Csurka et al, 2004]
[Vogel & Schiele, 2004]
[Jurie & Triggs, 2005]
[Lazebnik et al, 2006]
[Bosch et al, 2006]
Fisher Vector (FV) – Encoding

Dense set of local SIFT features → Fisher vector (high dim)

1\textsuperscript{st} order stats (k-th Gaussian): \( \Phi_k^{(1)} \sim \alpha_k \left( \frac{x - \mu_k}{\sigma_k} \right) \)

2\textsuperscript{nd} order stats (k-th Gaussian): \( \Phi_k^{(2)} \sim \alpha_k \left( \frac{(x - \mu_k)^2}{\sigma_k^2} - 1 \right) \)

stacking e.g. if SIFT x reduced to 80 dimensions by PCA

\[ \phi(x) = \left[ \Phi_1^{(1)}, \Phi_1^{(2)}, \ldots, \Phi_K^{(1)}, \Phi_K^{(2)} \right] \]

80-D 80-D 80-D

FV dimensionality: \( 80 \times 2 \times 512 = 81,920 \)

(for a mixture of 512 Gaussians)

Perronnin et al CVPR 07 & 10, ECCV 10
Projection Learning

Fisher vector (high dim) $\rightarrow$ low dimensional representation

$$ W \phi $$

- Learn projection onto a low-dim space where classes are well-separated
- Joint learning of projection $W$ and projected-space classifiers $\{v_c\}$ (WSABIE):

$$ \arg \min_{W,\{v_c\}} \sum_i \sum_{c' \neq c(i)} \max \left[ (v_{c'} - v_{c(i)})^T W \phi_i + 1, 0 \right] + \frac{\lambda}{2} \sum_c \|v_c\|_2^2 + \frac{\mu}{2} \|W\|_F^2 $$

- Or project onto the space of classifier scores: $W = \{w_c\}$
  - $\{w_c\}$ are linear SVM classifiers in the high-dimensional FV space
  - fast-to-learn
Deep Fisher Network

Input Image

0-th layer
- Dense feature extraction
  - SIFT, colour

1-st Fisher layer
- Spatial stacking
  - L2 norm. & PCA

2-nd Fisher layer
- (local & global pooling)
  - SSR & L2 norm.

Classifier layer
- One vs. rest linear SVMs

Shallow Fisher Vector

Dense feature extraction
- SIFT, raw patches, ...

SSR & L2 norm.

FV encoder
Fisher Layer

- L₂ norm-n & PCA decorrelation
- Spatial stacking (2×2)
- Compressed local Fisher encoding
Deep Fisher Network

input image

0-th layer

Dense feature extraction
SIFT, colour

Spatial stacking
low-dim FV encoder

L₂ norm. & PCA

1-st Fisher layer
(local & global pooling)

SSR & L₂ norm.
FV encoder

2-nd Fisher layer
(global pooling)

SSR & L₂ norm.
FV encoder

classifier layer

One vs. rest linear SVMs

One vs. rest linear SVMs

Dense feature extraction
SIFT, raw patches, ...

Shallow Fisher Vector
Classification Results for Fisher Network

ImageNet 2010 challenge dataset:
- 1.2M images, 1K classes
- SIFT & colour features
- Learning: 2-3 days on 200 CPU cores (MATLAB + MEX implementation)

<table>
<thead>
<tr>
<th>Method</th>
<th>top-1</th>
<th>top-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1\textsuperscript{st} FL (baseline)</td>
<td>55.4</td>
<td>76.4</td>
</tr>
<tr>
<td>2\textsuperscript{nd} FL</td>
<td>56.2</td>
<td>77.7</td>
</tr>
<tr>
<td>1\textsuperscript{st} and 2\textsuperscript{nd} FL</td>
<td>59.5</td>
<td>79.2</td>
</tr>
</tbody>
</table>

Improved classification accuracy by adding layer.
Deep ConvNet Implementation

• Based on cuda-convnet [Krizhevsky et al., 2012]
• 8 weight layers (rather narrow):
  conv64-conv256-conv256-conv256-conv256-full4096-full4096-full1000
• Jittering:
  • cropping, flipping, PCA-aligned noise
  • random occlusion:

• Single ConvNet instance
Classification Results

ImageNet 2012 challenge dataset:
- 1.2M images, 1K classes
- top-5 classification accuracy

<table>
<thead>
<tr>
<th>Method</th>
<th>top-5 accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>FV encoding (our 2012 entry)</td>
<td>72.7%</td>
</tr>
<tr>
<td><strong>Deep FishNet</strong></td>
<td><strong>76.9%</strong></td>
</tr>
<tr>
<td>Deep ConvNet [Krizhevsky et al., 2012]</td>
<td>81.8%</td>
</tr>
<tr>
<td></td>
<td>83.6% (5 ConvNets)</td>
</tr>
<tr>
<td>Deep ConvNet (our implementation)</td>
<td>82.3%</td>
</tr>
<tr>
<td><strong>Deep ConvNet + Deep FishNet</strong></td>
<td><strong>84.8%</strong></td>
</tr>
</tbody>
</table>

ConvNet and FisherNet are complementary
Outline

• Classification challenge
  • can Fisher Vector encodings be improved by a deep architecture?
  • deep Fisher Network (FN)
  • combination of two deep models: Convolutional Network (CN) and deep Fisher Network

• Localisation challenge
  • visualization of class saliency maps and per-image foreground pixels from a single classification CN
  • bounding boxes computed from foreground pixels
  • weak supervision: only image class labels used for training
Deep inside ConvNets: what Has Been Learnt?

ConvNet class model visualisation

- find a (regularised) image with a high class score $S_c(I, W)$:
  \[
  \arg\max_I S_c(I, W) - \lambda \|I\|^2_2
  \]
  with a fixed learnt model $W$
- compute $\partial S_c(I, W)/\partial I$ using back-prop

Cf ConvNet training

- max log-likelihood of the correct class
  \[
  \arg\max_W \sum_k \log P_{ck}(I_k, W)
  \]
- $\partial P_{ck}(I_k, W)/\partial W$ using back-prop

fox
pepper
dumbbell
Deep inside ConvNets: what Has Been Learnt?

ConvNet class model visualisation

- find a (regularised) image with a high class score $S_c(I, W)$:

$$\arg \max_I S_c(I, W) - \lambda \|I\|_2^2$$

with a fixed learnt model $W$

- compute $\partial S_c(I, W)/\partial I$ using back-prop

NB $\arg \max_I P_c(I, W)$

gives less prominent visualisation, as it concentrates on reducing scores of other classes

Deep inside ConvNets: What Makes an Image Belong to a Class?

• ConvNets are highly non-linear $\rightarrow$ local linear approximation

• 1st order expansion of a class score around a given image $I_0$:

$$S_c(I_0, w) \approx w^T I + b$$  – score of $c$-th class

$$w = \frac{\partial S_c}{\partial I} \bigg|_{I_0}$$  – computed using back-prop

• $w$ has the same dimensions as image $I_0$

• magnitude of $w$ defines a saliency map for image $I_0$ and class $c$

Saliency Maps For Top-1 Class
Saliency Maps For Top-1 Class
Saliency Maps For Top-1 Class
Image Saliency Map

- **Weakly supervised**
  - computed using class-n ConvNet, trained on image class labels
  - no additional annotation required (e.g. boxes or masks)
- Highlights discriminative object parts
- Instant computation – no sliding window
- Fires on several object instances

- Related to deconvnet [Zeiler and Fergus, 2013]
  - very similar for convolution, max-pooling, and RELU layers
  - but we also back-prop through fully-connected layers
Saliency Maps for Object Localisation

- Image $\rightarrow$ top-k class $\rightarrow$ class saliency map $\rightarrow$ object box
BBox Localisation for ILSVRC Submission

- Given an image and a saliency map:
BBox Localisation for ILSVRC Submission

- Given an image and a saliency map:
  1. Foreground/background mask using thresholds on saliency

blue – foreground
cyan – background
red – undefined
BBox Localisation for ILSVRC Submission

- Given an image and a saliency map:
  1. Foreground/background mask using thresholds on saliency
  2. GraphCut colour segmentation
     [Boykov and Jolly, 2001]
BBox Localisation for ILSVRC Submission

• Given an image and a saliency map:
  1. Foreground/background mask using thresholds on saliency
  2. GraphCut colour segmentation [Boykov and Jolly, 2001]
  3. Bounding box of the largest connected component

• Colour information propagates segmentation from the most discriminative areas
Segmentation-Localisation Examples
Segmentation-Localisation Examples
Segmentation-Localisation Failure Cases

- Several object instances
Segmentation-Localisation Failure Cases

- Segmentation isn’t propagated from the salient parts
Segmentation-Localisation Failure Cases

• Limitations of GraphCut segmentation
Summary

- Fisher encoding benefits from stacking
- Deep FishNet is complementary to Deep ConvNet
- Class saliency maps are useful for localisation
  - location of discriminative object parts
  - weakly supervised: bounding boxes not used for training
  - fast to compute