Adaptive Non-parametric Rectification of Shallow and Deep Experts

Min LIN*, Qiang CHEN*, Jian DONG, Junshi HUANG, Wei XIA

Shuicheng YAN

eleyans@nus.edu.sg

National University of Singapore

(* indicates equal contribution)
Task 2: Classification – NUS Solution Overview

ILSVRC 2013 Dataset

Super-coding

Shallow Experts PASCAL VOC 2012 Solution (SVMs)

Deep Experts Convolutional Neural Network

Adaptive Non-parametric Rectification

Finished

Unfinished due to surgery of key member, but effective

Bigger and Deeper

“Network in Network” NIN: CNN with Non-linear Filters, yet No Final Fully-connected NN Layer

2/15
Non-parametric Rectification

- **Motivation**
  - Each validation-set image has a **pair of outputs-from-experts** \((x_i)\) and **ground-truth label** \((y_i)\), possibly inconsistent
  - For a testing image, rectify the experts based on priors from validation-set pairs (**experts errors are often repeated**)

\[
x = \begin{bmatrix}
\end{bmatrix}
\Rightarrow
y = \begin{bmatrix}
\end{bmatrix}
\]

**Label propagation by affinities**

Finally, the prediction is rectified as \(\alpha x + (1 - \alpha)y\)
Adaptive Non-parametric Rectification

- Determine the optimal tuneable values for each test sample
  - For each test sample, refer to the k-NN in the validation set
  - Optimal tuneable values for validation samples are obtained through cross-validation
Shallow Experts

Two-layer feature representation

- **Layer 1**: Traditional handcrafted features
  - We exact dense-SIFT, HOG and color moment features within patches

- **Layer 2**: Coding + Pooling
  - Derivative coding: Fisher-Vector
  - Parametric coding: Super-Coding

5/15
Shallow Experts: GMM-based **Super-Coding**

- Two basic strategies to obtain the patch based GMM coding [1]
  - **Derivative**: Fisher-Vector (w.r.t. $\mu_i$ and $\sigma_i$, high-order), Super-Vector (w.r.t. $\mu_i$ only)
  
  \[ g_{\mu_i}^X = \frac{1}{T \sqrt{\omega_i}} \sum_{i=1}^{T} \gamma(i) \left( \frac{x_i - \mu_i}{\sigma_i} \right) \]
  
  \[ g_{\sigma_i}^X = \frac{1}{T \sqrt{2 \omega_i}} \sum_{i=1}^{T} \gamma(i) \left[ \frac{(x_i - \mu_i)^2}{\sigma_i^2} - 1 \right] \]

  Image from [F Perronnin, 2012]

- **Parametric**: use adapted model parameters, e.g. Mean-Vector (1\(^{st}\) order)

- High-order parametric coding
  - The **Super-Coding**: $C_a = [\frac{\mu_1}{\sqrt{\sigma_1}}; \cdots; \frac{\mu_K}{\sqrt{\sigma_K}}; \frac{\sigma_1}{\sigma_1}; \cdots; \frac{\sigma_K}{\sigma_K}]$

  The inner product of the codings is an approximate of the KL-divergence

- Advantages
  - Comparable and **complementary** performance with Fisher-Vector
  - It is very efficient to compute Super-Coding along with Fisher-Vector

---

Shallow Experts: **Early-stop SVMs**

- **Two-layer feature representation**
  - **Layer 1:** Traditional handcrafted features
    - We use dense-SIFT, HOG and color moment
  - **Layer 2:** Coding + Pooling
    - Derivative coding: Fisher-Vector
    - Parametric coding: Super-Coding

- **Classifier learning**
  - Dual coordinate descent SVM [2]
  - Model averaging for early stopped SVMs

---

Shallow Experts: Performance

- Results on validation set
  - 1024-component GMM
  - Average early-stopped SVMs
    - For each round, 1) randomly select 1/10 of the negative samples, and 2) stop the SVMs at around 30 epochs \textbf{[balance efficiency and performance]}
    - Train 3 rounds, and average

<table>
<thead>
<tr>
<th></th>
<th>Fisher-Vector (FV)</th>
<th>Super-Coding (SC)</th>
<th>FV+SC</th>
<th>3 FV+SC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 1</td>
<td>47.93%</td>
<td>47.67%</td>
<td>45.3%</td>
<td>43.27%</td>
</tr>
<tr>
<td>Top 5</td>
<td>25.93%</td>
<td>25.54%</td>
<td>24.0%</td>
<td>22.5%</td>
</tr>
</tbody>
</table>

Comparable & complementary
Deep Experts

Follow Krizhevsky et al. [3]

- Achieved top-1 performance 1% better than reported by Krizhevsky
- No network splitting for two GPUs, instead NVIDIA TITAN GPU card 6GB memory
- Our network does not have PCA noise for data expansion, which is reported by Krizhevsky to improve the performance by 1%

<table>
<thead>
<tr>
<th></th>
<th>Krizhevsky’s</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 1</td>
<td>40.7%</td>
<td>39.7%</td>
</tr>
<tr>
<td>Top 5</td>
<td>18.2%</td>
<td>17.8%</td>
</tr>
</tbody>
</table>
Deep Experts: Extensions

- Two extensions

- Bigger (left): Big network with doubled convolutional filters/kernels
- Deeper (right): CNN with 6 convolutional layers

Performance comparison on validation set

<table>
<thead>
<tr>
<th></th>
<th>CNN5 (8days)</th>
<th>BigNet (30days)</th>
<th>CNN6 (12days)</th>
<th>5 CNN6</th>
<th>5 CNN6 +BigNet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 1</td>
<td>39.7%</td>
<td>37.67%</td>
<td>38.32%</td>
<td>36.27%</td>
<td>35.96%</td>
</tr>
<tr>
<td>Top 5</td>
<td>17.8%</td>
<td>15.96%</td>
<td>16.52%</td>
<td>15.21%</td>
<td>14.95%</td>
</tr>
</tbody>
</table>

10/15
Deep Experts: “Network in Network” (NIN)

- **NIN**: CNN with non-linear filters, yet without final fully-connected NN layer
Deep Experts: **“Network in Network” (NIN)**

- **NIN**: CNN with non-linear filters, yet without final fully-connected NN layer

- Intuitively less overfitting globally, and more discriminative locally

  *(not finally used in our submission due to the surgery of our main team member, but very effective)*

<table>
<thead>
<tr>
<th></th>
<th>Cifar-10</th>
<th>Cifar-100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Previous Best performance (Maxout) [4]</td>
<td>11.68%</td>
<td>38.57%</td>
</tr>
<tr>
<td>Our method</td>
<td>10.41%</td>
<td>36.30%</td>
</tr>
</tbody>
</table>


## NUS Submissions

- **Results on test set**

<table>
<thead>
<tr>
<th>Submission</th>
<th>Method</th>
<th>Top 5 error rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>tf</td>
<td>traditional framework based on PASCAL VOC12 winning solution with extension of high-order parametric coding</td>
<td>22.39% (26.17%)</td>
</tr>
<tr>
<td>cnn</td>
<td>weighted sum of outputs from one large CNN and five CNNs with 6-convolutional layers</td>
<td>15.02% (16.42%)</td>
</tr>
<tr>
<td>weightt tune</td>
<td>weighted sum of all outputs from CNNs and refined PASCAL VOC12 winning solution</td>
<td>13.98% (↓1.04%)</td>
</tr>
<tr>
<td>anpr</td>
<td>adaptive non-parametric rectification of all outputs from CNNs and refined PASCAL VOC12 winning solution</td>
<td>13.30% (↓0.68%)</td>
</tr>
<tr>
<td>anpr retrain</td>
<td>adaptive non-parametric rectification of all outputs from CNNs and refined PASCAL VOC12 winning solution, with further CNN retraining on the validation set</td>
<td>12.95% (↓0.35%)</td>
</tr>
</tbody>
</table>

Clarifai 11.74% (↓1.21%)

---

13/15
Conclusions & Further Work

- **Conclusions**
  - Complementarity of shallow and deep experts
  - Super-coding: effective, complementary with Fisher-Vector
  - Deep learning: deeper & bigger, better

- **Further work**
  - Consider more validation data for adaptive non-parametric rectification
    (training data are overfit, yet only 50k validation data; training: less is more)
  - Network in Network (NIN): CNN with non-linear filters, yet without final fully-connected NN layer on ILSVRC data; paper draft is accessible at http://arxiv.org/abs/1312.4400
Thank You!

Shuicheng YAN

eleyans@nus.edu.sg