Improving Object Detection with Deep Convolutional Networks via Bayesian Optimization and Structured Prediction

Yuting Zhang*, Kihyuk Sohn†, Ruben Villegas†, Gang Pan*, Honglak Lee†
Object detection using deep learning

- **Object detection systems** based on the **deep convolutional neural network (CNN)** have recently made ground-breaking advances.
  
  [LeCune et al. 1989; Sermanet et al. 2013; Girschick et al., 2014; Simoyan et al., 2014; Lin et al. 2014, and many others]

- **State-of-the-art**: “Regions with CNN features” (R-CNN)
  

Image adapted from Girshick et al., 2014
R-CNN: Method

1) Convolutional neural network for classification

- Pretrained on ImageNet for 1000-category classification
- Finetuned on PASCAL VOC for 20 categories


2) Selective search for region proposal:

- Hierarchical segmentation → bounding box


Images from Krizhevsky et al. 2012 & Sande et al. 2011
R-CNN: Detection

Classification confidence for sampled bounding boxes

Detection: locally solve

$$\arg\max_y f(x, y)$$

where $x$ is the image, and $y$ is a bounding box, $f(x, y)$ is the classification confidence computed from CNN.


R-CNN: Pros and Cons

Pros:

• Surprisingly good performance (mean average precision, mAP), e.g., on PASCL VOC2007:
  
  • Deformable part model (old SOA): 33.4%
  • R-CNN: 53.7%

• Strong discriminative ability from CNN
• Reasonable efficiency from region proposal
R-CNN: Pros and Cons

Pros:
• Surprisingly good performance (mean average precision, mAP), e.g., on PASCL VOC2007:
  • Deformable part model (old SOA): 33.4%
  • R-CNN: 53.7%
• Strong discriminative ability from CNN
• Reasonable efficiency from region proposal

Cons:
• Poor localization (worse than DPM), due to
  • Ground truth bounding box (BBox) may be missing from (or have poor overlap with) region proposals
  • CNN is trained solely for classification, but not localization
Our solutions

1. Find better bounding boxes via **Bayesian optimization**
2. Improve localization sensitivity via **structured objective**
Thrust 1: Find better bounding boxes via Bayesian optimization
Fine-grained search: Framework
Given a test image
Propose initial regions via selective search
Compute classification scores

Detection score $f(x,y_{1:N};w)$

CNN-based Classifier
What if no existing bounding box is good enough?

How to propose a better box?
Find a local optimal bounding box

Local optimum

CNN-based Classifier

Detection score $f(x,y_{1:N};w)$
Determine a local search region

Search Region near local optimum for Bayesian optimization
Propose a bounding box via Bayesian optimization

Search Region near local optimum for Bayesian optimization

The new box has a good chance to get better classification score
Compute the actual classification score
Iterative procedure: Iteration 2
Iteration 2: Find a local optimum
Iteration 2: Determine a local search region

Search Region near local optimum for Bayesian optimization
Iteration 2: Propose a new box via Bayesian opt.

Search Region near local optimum for Bayesian optimization
Iteration 2: compute the actual score
After a few iterations ...
Final detection output

Pruned by threshold

Before NMS

After NMS
Bayesian optimization: General

e.g., CNN-based classifier or any score function of detection methods.

• Model the \textbf{complicated function} \( f(x, y) \), whose evaluation cost is high, with a \textbf{probabilistic distribution of function values}.

• The distribution is defined with a \textbf{relatively computationally efficient} surrogate model.

Framework

• Let \( \mathcal{D}_N = \{y_j, f_j\}_{j=1}^N \) and \( f_j = f(x, y_j) \) be the known solutions. We want to model

\[
p(f|\mathcal{D}_N) \propto p(\mathcal{D}_N|f)p(f)
\]

• Try to find a new boxing box \( y_{N+1} \neq y_j, \forall j \leq N \) with the highest chance s.t. \( f_{N+1} > \max_{1 \leq j \leq N} f_j \)
Bayesian optimization: Gaussian process

- Framework:
  \[ p(f | D_N) \propto p(D_N | f)p(f) \]

- Gaussian process is a general function prior, which used for \( p(f) \).
- \( p(f_{N+1} | y_{N+1}, D_N) \) can be expressed as a multivariate Gaussian, whose parameters can be obtained by Gaussian process regression (GPR) as a closed-form solution, when the square exponential covariance function is used.
- The chance of \( f_{N+1} > \max_{1 \leq j \leq N} f_j = \hat{f}_N \)
  is measure by the expected improvement:
  \[ \int_{\hat{f}_N}^f (f - \hat{f}_N) \cdot p(f | y_{N+1}, D_N; \theta) df \]
FGS Procedure: a real example
Original image

The image is from PASCAL VOC2007
Initial region proposals
Initial detection (local optima)
Initial detection & Ground truth

Take this as ONE starting point

Neither gives good localization
Iter 1: Boxes inside the local search region
Iter 1: Heat map of expected improvement (EI)

- A box has 4-coordinates: (centerX, centerY, height, width)
- The height and width are normalized by max to visualize EI in 2D
Iter 1: Heat map of expected improvement (EI)
Iter 1: Maximum of EI – the newly proposed box
Iter 1: Complete
Iteration 2: local optimum & search region
Iteration 2: EI heat map & new proposal
Iteration 2: Newly proposed box & its actual score
Iteration 3: local optimum & search region
Iteration 3: EI heat map & new proposal
Iteration 3: Newly proposed box & its actual score
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Final results
Final results & Ground truth
Thrust 2: 
Train CNN classifier with structured output regression
Structured loss for detection

- Linear classifier
  \[ g(x; w) = \arg\max_{y \in Y} f(x, y; w) \]
  \[ f(x, y; w) = w^T \tilde{\phi}(x, y) \]

  \[ \tilde{\phi}(x, y) = \begin{cases} 
  \phi(x, y), & l = +1 \\
  0, & l = -1 
  \end{cases} \]

- Minimizing the structured loss (Blaschko and Lampert, 2008)*

  \[ \hat{w} = \arg\max_w \sum_{i=1}^{M} \Delta(g(x_i; w), y_i) \]

  \[ \Delta(y, y_i) = \begin{cases} 
  1 - \text{IoU}(y, y_i), & \text{if } l = l_i = 1 \\
  0, & \text{if } l = l_i = -1 \\
  1, & \text{if } l \neq l_i 
  \end{cases} \]


Other related work: LeCun et al. 1989; Taskar et al. 2005; Joachims et al. 2005; Veldaldi et al. 2014; Thomson et al. 2014; and many others
Structured SVM for detection

• The objective is hard to solve. Replace it with an upper-bound surrogate using structured SVM framework

\[
\min_\mathbf{w} \quad \frac{1}{2} \| \mathbf{w} \|^2 + \frac{C}{M} \sum_{i=1}^{M} \xi_i, \text{ subject to}
\]

\[
\mathbf{w}^T \tilde{\phi}(x_i, y_i) \geq \mathbf{w}^T \tilde{\phi}(x_i, y) + \Delta(y, y_i) - \xi_i, \forall y \in \mathcal{Y}, \forall i
\]

\[
\xi_i \geq 0, \forall i
\]

• The constraints can be re-written as:

\[
\mathbf{w}^T \phi(x_i, y_i) \geq 1 - \xi_i, \quad \forall i \in I_{pos},
\]

\[
\mathbf{w}^T \phi(x_i, y) \leq -1 + \xi_i, \quad \forall y \in \mathcal{Y}, \forall i \in I_{neg},
\]

\[
\mathbf{w}^T \phi(x_i, y_i) \geq \mathbf{w}^T \phi(x_i, y) + \Delta^{loc}(y, y_i) - \xi_i,
\]

\[
\forall y \in \mathcal{Y}, \forall i \in I_{pos},
\]

where \( \Delta^{loc}(y, y_i) = 1 - \text{IoU}(y, y_i) \).
Solution for Structured SVM

• Approximate the structured output space $\mathcal{Y}$ with samples from selective search and random boxes near ground truths.

• Gradient-based method
  • Opt 1: LBFG-S for learning classification layer
  • Opt 2: SGD for fine-tuning the whole CNN

• Hard sample mining according to hinge loss
  • Not all the training samples can fit into memory
  • Significantly reduce the time consumption for searching the most violated sample
Experimental results
Control experiments with Oracle detector

- Oracle detector for image $x_i$, and ground truth box $y_i$

$$f_{\text{ideal}}(x_i, y) = \text{IoU}(y, y_i)$$

where IoU is the intersection over union.
Controlled experiments with Oracle detector

More region proposal methods:

- SS: selective search
  - fast (default) / extended / quality
- Objectness*
- Local random search:
  - Random generate extra boxes without Bayesian optimization

Controlled experiments with Oracle detector

- x-axis: Different IoU thresholds for accepting a true positive
- y-axis: mean average precision (mAP)
Control experiments with Oracle detector

More region proposal methods:
• SS: selective search
  fast (default) / extended / quality
• Objectness

Results:
• x-axis: Different IoU thresholds for accepting a true positive
• y-axis: mean average precision (mAP)

- Local random search:
  Random generate extra boxes without Bayesian optimization

More region proposal methods:

SS (~2000 boxes per image)
- SS + Local random search (~2100 boxes per image)
- SS + FGS (~2100 boxes per image)
- SS + Objectness (~3000 boxes per image)
- SS extended (~3500 boxes per image)
- SS quality (~10000 boxes per image)

mAP / %

IoU threshold for true positives

0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9

0 10 20 30 40 50 60 70 80 90 100
FGS efficiency: time overhead

- Baseline time: Initial feature extraction time of R-CNN

![Graph showing actual time consumption over FTS iteration number.
- Red line: Feature extraction.
- Blue line: GP regression, etc.
- X-axis: Maximum FTS iteration number ($t_{max}$).
- Y-axis: Actual time consumption (ratio).
- Y-values: 0 (0%), 5 (3%), 10 (6%), 15 (9%), 20 (13%), 25 (16%).]
mAP on VOC2007 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>Standard localization</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN (AlexNet)</td>
<td>58.5</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>65.4</td>
</tr>
</tbody>
</table>

Bounding box regression is always taken as a post-processing step.
### mAP on VOC2007 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>Standard localization</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN (AlexNet)</td>
<td>58.5</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>65.4</td>
</tr>
<tr>
<td>+ StructObj</td>
<td>66.6</td>
</tr>
<tr>
<td>+ StructObj-FT</td>
<td>66.9</td>
</tr>
</tbody>
</table>

mAP on VOC2007 test set is 1.2%
### mAP on VOC2007 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>Standard localization</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN (AlexNet)</td>
<td>58.5</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>65.4</td>
</tr>
<tr>
<td>+ StructObj</td>
<td>66.6</td>
</tr>
<tr>
<td>+ StructObj-FT</td>
<td>66.9</td>
</tr>
<tr>
<td>+ FGS</td>
<td>67.2</td>
</tr>
</tbody>
</table>
mAP on VOC2007 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>Standard localization</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN (AlexNet)</td>
<td>58.5</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>65.4</td>
</tr>
<tr>
<td>+ StructObj</td>
<td>66.6</td>
</tr>
<tr>
<td>+ StructObj-FT</td>
<td>66.9</td>
</tr>
<tr>
<td>+ FGS</td>
<td>67.2</td>
</tr>
<tr>
<td>+ FGS + StructObj</td>
<td>68.5</td>
</tr>
<tr>
<td>+ FGS + StructObj-FT</td>
<td>68.4</td>
</tr>
</tbody>
</table>
### mAP on VOC2007 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>IoU&gt;0.5</th>
<th>IoU&gt;0.7</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN (AlexNet)</td>
<td>58.5</td>
<td></td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>65.4</td>
<td></td>
</tr>
<tr>
<td>+ StructObj</td>
<td>66.6</td>
<td></td>
</tr>
<tr>
<td>+ StructObj-FT</td>
<td>66.9</td>
<td></td>
</tr>
<tr>
<td>+ FGS</td>
<td>67.2</td>
<td></td>
</tr>
<tr>
<td>+ FGS + StructObj</td>
<td>68.5</td>
<td></td>
</tr>
<tr>
<td>+ FGS + StructObj-FT</td>
<td>68.4</td>
<td></td>
</tr>
</tbody>
</table>
mAP on VOC2007 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>IoU&gt;0.5</th>
<th>IoU&gt;0.7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Standard localization</td>
<td>More accurate localization</td>
</tr>
<tr>
<td>R-CNN (AlexNet)</td>
<td>58.5</td>
<td>35.2</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>65.4</td>
<td>35.2</td>
</tr>
<tr>
<td>+ StructObj</td>
<td>66.6</td>
<td>40.5</td>
</tr>
<tr>
<td>+ StructObj-FT</td>
<td>66.9</td>
<td>41.8</td>
</tr>
<tr>
<td>+ FGS</td>
<td>67.2</td>
<td>42.7</td>
</tr>
<tr>
<td>+ FGS + StructObj</td>
<td>68.5</td>
<td>43.0</td>
</tr>
<tr>
<td>+ FGS + StructObj-FT</td>
<td>68.4</td>
<td>43.7</td>
</tr>
</tbody>
</table>
## mAP on VOC2007 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>IoU &gt; 0.5</th>
<th>IoU &gt; 0.7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Standard localization</td>
<td>More accurate localization</td>
</tr>
<tr>
<td>R-CNN (AlexNet)</td>
<td>58.5</td>
<td>35.2</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>65.4</td>
<td>35.2</td>
</tr>
<tr>
<td>+ StructObj</td>
<td>66.6</td>
<td>40.5</td>
</tr>
<tr>
<td>+ StructObj-FT</td>
<td>66.9</td>
<td>41.8</td>
</tr>
<tr>
<td>+ FGS</td>
<td>67.2</td>
<td>42.7</td>
</tr>
<tr>
<td>+ FGS + StructObj</td>
<td><strong>68.5</strong></td>
<td>43.0</td>
</tr>
<tr>
<td>+ FGS + StructObj-FT</td>
<td>68.4</td>
<td><strong>43.7</strong></td>
</tr>
</tbody>
</table>
### mAP on VOC2007 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>IoU&gt;0.5 Standard localization</th>
<th>IoU&gt;0.7 More accurate localization</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN (AlexNet)</td>
<td>58.5</td>
<td>35.2</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>65.4</td>
<td>35.2</td>
</tr>
<tr>
<td>+ StructObj</td>
<td>66.6</td>
<td>40.5</td>
</tr>
<tr>
<td>+ StructObj-FT</td>
<td>66.9</td>
<td>41.8</td>
</tr>
<tr>
<td>+ FGS</td>
<td>67.2</td>
<td>42.7</td>
</tr>
<tr>
<td>+ FGS + StructObj</td>
<td><strong>68.5</strong></td>
<td>43.0</td>
</tr>
<tr>
<td>+ FGS + StructObj-FT</td>
<td>68.4</td>
<td><strong>43.7</strong></td>
</tr>
</tbody>
</table>
mAP on VOC2012 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>IoU&gt;0.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN (AlexNet)</td>
<td>53.3</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>63.0</td>
</tr>
<tr>
<td>+ StructObj</td>
<td>65.1</td>
</tr>
<tr>
<td>+ FGS</td>
<td>64.0</td>
</tr>
<tr>
<td>+ FGS + StructObj</td>
<td>66.4</td>
</tr>
</tbody>
</table>
### mAP on VOC2012 test set

<table>
<thead>
<tr>
<th>Mean Average Precision</th>
<th>IoU&gt;0.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN (AlexNet)</td>
<td>53.3</td>
</tr>
<tr>
<td>R-CNN (VGGNet)</td>
<td>63.0</td>
</tr>
<tr>
<td>+ StructObj</td>
<td>65.1</td>
</tr>
<tr>
<td>+ FGS</td>
<td>64.0</td>
</tr>
<tr>
<td>+ FGS + StructObj</td>
<td>66.4</td>
</tr>
<tr>
<td>Network in Network*</td>
<td>63.8</td>
</tr>
</tbody>
</table>

*2.6% improvement from Network in Network*

---

*M. Lin, Q. Chen, S. Yan, Network In Network, ICLR 2014*
Good examples on VOC 2007 (1)

aeroplane  bycycle  bird  boat  bottle

bus  car  cat  chair  cow

diningtable  dog  horse  motorbike  person

pottedplant  sheep  sofa  train  tvmonitor
Good examples on VOC 2007 (1)

Red boxes: R-CNN (VGGNet) baseline.
Good examples on VOC 2007 (1)

Red boxes: R-CNN (VGGNet) baseline.

Green boxes: Ground truth (GT)
Good examples on VOC 2007 (1)

Numbers: 
Overlap (IoU) with GT

Red boxes: 
R-CNN (VGGNet) baseline.

Green boxes: 
Ground truth (GT)
Good examples on VOC 2007

**Numbers:**
Overlap (IoU) with GT

**Red boxes:**
R-CNN (VGGNet) baseline.

**Green boxes:**
Ground truth (GT)

**Yellow boxes:**
Ours (+ StructObj + FGS)
Good examples on VOC 2007 (2)

aeroplane  bycycle  bird  boat  bottle

bus  car  cat  chair  cow
diningtable  dog  horse  moterbike  person

pottedplant  sheep  sofa  train  tvmonitor
Good examples on VOC 2007 (2)

Red boxes: R-CNN (VGGNet) baseline.
Good examples on VOC 2007 (2)

Red boxes: R-CNN (VGGNet) baseline.

Green boxes: Ground truth (GT)
Good examples on VOC 2007 (2)

Numbers:
Overlap (IoU) with GT

Red boxes:
R-CNN (VGGNet) baseline.

Green boxes:
Ground truth (GT)
Good examples on VOC 2007 (2)

Numbers:
Overlap (IoU) with GT

Red boxes:
R-CNN (VGGNet) baseline.

Green boxes:
Ground truth (GT)

Yellow boxes:
Ours (+ StructObj + FGS)
Conclusion

• We proposed two complementary methods for improving object detection
  1. Find better bounding boxes via Bayesian optimization
  2. Improve localization sensitivity via structured objective

• If the object classifier is accurate, our fine-grained search algorithm is almost as good as doing exhaustive search.
  • compatible with most detection methods.

• We significantly improve over the previous state-of-the-art in object detection both for VOC 2007 and 2012 benchmarks.
Code available at:

bit.ly/fgs-obj

Q & A

Thank you!
References


