Evaluating Weakly-Supervised Object Localization Methods Right

Junsuk Choe
Yonsei University

Seong Joon Oh
Clova AI Research NAVER Corp.

Seungho Lee
Yonsei University

Sanghyuk Chun
Clova AI Research NAVER Corp.

Zeynep Akata
University of Tübingen

Hyunjung Shim
Yonsei University

* Equal contribution
What is the paper about?

Weakly-supervised object localization methods have many issues.

E.g. they are often not truly "weakly-supervised".

We fix the issues.
Weakly-supervised object localization?
Classification

Object localization

Semantic segmentation

Instance segmentation

What's in the image?

Where's the cat?

Classify each pixel in image:

Classify pixels by instance:
What's in the image?

Classify each pixel in image:

A: Cat

Where's the cat?

Object localization
Classification vs. Semantic Segmentation:

**Classification**
- A: Cat

**Semantic Segmentation**
- Classify each pixel in image:
- Where's the cat?
  - The image **must** contain a single class.
  - The class is known.
  - FG-BG mask as final output.

**Object Localization**
- Instance segmentation
Task goal: FG-BG mask
Supervision types

Task goal: FG-BG mask

Weak supervision: Class label
Full supervision: FG-BG mask
Strong supervision: Part parsing mask
Supervision types

- **Full supervision:** FG-BG mask
- **Strong supervision:** Part parsing mask
- **Weak supervision:** Image-level class labels are examples of weak supervision for localization task.
Weakly-supervised object localization

Test-time task: Localization.

Input image → FG-BG mask

Train-time supervision: Images + class labels

Input image + Cat
How to train a WSOL model.
CAM example (CVPR'16)
How to train a WSOL model. CAM example (CVPR'16)
CAM at test time.

Input image → Model → Score map → Thresholding → FG-BG mask
We didn't used any full supervision, did we?
Implicit full supervision for WSOL.

Which threshold do we choose?
Implicit full supervision for WSOL.

Validation localization: 74.3%

Threshold 0.25

Validation set GT mask
Implicit full supervision for WSOL.

Validation set GT mask

Validation localization: 74.3%

Threshold 0.25 → 0.30

"Try different threshold"
Implicit full supervision for WSOL.

Validation localization: 74.3% → 82.9%

Threshold 0.25 → 0.30

"Try different threshold"
WSOL methods have many hyperparameters to tune.

<table>
<thead>
<tr>
<th>Method</th>
<th>Hyperparameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAM, CVPR'16</td>
<td>Threshold / Learning rate / Feature map size</td>
</tr>
<tr>
<td>HaS, ICCV'17</td>
<td>Threshold / Learning rate / Feature map size / Drop rate / Drop area</td>
</tr>
<tr>
<td>ACoL, CVPR'18</td>
<td>Threshold / Learning rate / Feature map size / Erasing threshold</td>
</tr>
<tr>
<td>SPG, ECCV'18</td>
<td>Threshold / Learning rate / Feature map size / Threshold 1L / Threshold 1U / Threshold 2L / Threshold 2U / Threshold 3L / Threshold 3U</td>
</tr>
<tr>
<td>ADL, CVPR'19</td>
<td>Threshold / Learning rate / Feature map size / Drop rate / Erasing threshold</td>
</tr>
<tr>
<td>CutMix, ICCV'19</td>
<td>Threshold / Learning rate / Feature map size / Size prior / Mix rate</td>
</tr>
</tbody>
</table>

• Far more than usual classification training.
Hyperparameters are often searched through validation on full supervision.

- [...] the thresholds were chosen by observing a few qualitative results on training data. HaS, ICCV'17.

- The thresholds [...] are adjusted to the optimal values using grid search method. SPG, ECCV'18.

- Other methods do not reveal the selection mechanism.
This practice is against the philosophy of WSOL.
But we show in the following that the full supervision is inevitable.
WSOL is ill-posed without full supervision.

Pathological case:

A class (e.g. duck) correlates better with a BG concept (e.g. water) than a FG concept (e.g. feet).

Then, WSOL is not solvable.

See Lemma 3.1 in paper.
So, let's use full supervision.
But in a controlled manner.
Do the **validation** explicitly, but with the **same** data.

For each WSOL benchmark dataset, define splits as follows.

- **Training**: Weak supervision for model training.
- **Validation**: Full supervision for hyperparameter search.
- **Test**: Full supervision for reporting final performance.
Existing benchmarks did not have the validation split.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training set (Weak sup)</th>
<th>Validation set (Full sup)</th>
<th>Test set (Full sup)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageNet</td>
<td>✓</td>
<td>✗ (ImageNetV2[a] exists, but no full sup.)</td>
<td>✓</td>
</tr>
<tr>
<td>CUB</td>
<td>✓</td>
<td>✗ (No images, nothing.)</td>
<td>✓</td>
</tr>
</tbody>
</table>

Our benchmark proposal.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training set (Weak sup)</th>
<th>Validation set (Full sup)</th>
<th>Test set (Full sup)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageNet</td>
<td>✓</td>
<td>✓ ImageNetV2 + Our annotations.</td>
<td>✓</td>
</tr>
<tr>
<td>CUB</td>
<td>✓</td>
<td>✓ Our image collections + Our annotations.</td>
<td>✓</td>
</tr>
</tbody>
</table>
Our benchmark proposal.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training set (Weak sup)</th>
<th>Validation set (Full sup)</th>
<th>Test set (Full sup)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageNet</td>
<td>✓</td>
<td>✓ ImageNetV2 + Our annotations.</td>
<td>✓</td>
</tr>
<tr>
<td>CUB Newly introduced dataset.</td>
<td>✓</td>
<td>✓ Our image collections + Our annotations.</td>
<td>✓</td>
</tr>
</tbody>
</table>
Do the validation explicitly, with the same search algorithm.

For each WSOL method, tune hyperparameters with

- Optimization algorithm: Random search.
- Search space: Feasible range (not "reasonable range").
- Search iteration: 30 tries.
Do the **validation** explicitly, with the **same** search algorithm.

<table>
<thead>
<tr>
<th>Method</th>
<th>Hyperparameters</th>
<th>Search space (Feasible range)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAM, CVPR'16</td>
<td>Learning rate, Feature map size</td>
<td>LogUniform[0.00001, 1] Categorical{14, 28}</td>
</tr>
<tr>
<td>HaS, ICCV'17</td>
<td>Learning rate, Feature map size, Drop rate, Drop area</td>
<td>LogUniform[0.00001, 1] Categorical{14, 28} Uniform[0, 1] Uniform[0, 1]</td>
</tr>
<tr>
<td>ACoL, CVPR'18</td>
<td>Learning rate, Feature map size, Erasing threshold</td>
<td>LogUniform[0.00001, 1] Categorical{14, 28} Uniform[0, 1]</td>
</tr>
<tr>
<td>SPG, ECCV'18</td>
<td>Learning rate, Feature map size, Threshold 1L, Threshold 1U, Threshold 2L, Threshold 2U</td>
<td>LogUniform[0.00001, 1] Categorical{14, 28} Uniform[0, 1] Uniform[d1, 1] Uniform[0, 1]</td>
</tr>
<tr>
<td>ADL, CVPR'19</td>
<td>Learning rate, Feature map size, Drop rate, Erasing threshold</td>
<td>LogUniform[0.00001, 1] Categorical{14, 28} Uniform[0, 1] Uniform[0, 1]</td>
</tr>
<tr>
<td>CutMix, ICCV'19</td>
<td>Learning rate, Feature map size, Size prior, Mix rate</td>
<td>LogUniform[0.00001, 1] Categorical{14, 28} 1/Uniform[0.2]-1/2 Uniform[0, 1]</td>
</tr>
</tbody>
</table>
Previous treatment of the score map threshold.
Previous treatment of the score map threshold.

- Score maps are natural outputs of WSOL methods.
- The binarizing threshold is sometimes tuned, sometimes set as a "common" value.
But setting the right threshold is critical.
But setting the right threshold is critical.

- Method 1 seems to perform better: it covers the object extent better.
But setting the right threshold is critical.

- But at the method-specific optimal threshold, Method 2 (62.8 IoU) > Method 1 (61.2 IoU).
We propose to remove the threshold dependence.

- **MaxBoxAcc**: For box GT, report accuracy at the best score map threshold.

  ★ **Max** performance over score map thresholds.

- **PxAP**: For mask GT, report the AUC for the pixel-wise precision-recall curve parametrized by the score map threshold.

  ★ **Average** performance over score map thresholds.
Remaining issues for fair comparison.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>ImageNet</th>
<th>CUB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VGG</td>
<td>Inception</td>
</tr>
<tr>
<td>CAM ’16</td>
<td>42.8</td>
<td>-</td>
</tr>
<tr>
<td>HaS ’17</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>ACoL ’18</td>
<td>45.8</td>
<td>-</td>
</tr>
<tr>
<td>SPG ’18</td>
<td>-</td>
<td>48.6</td>
</tr>
<tr>
<td>ADL ’19</td>
<td>44.9</td>
<td>48.7</td>
</tr>
<tr>
<td>CutMix ’19</td>
<td>43.5</td>
<td>-</td>
</tr>
</tbody>
</table>

- Different datasets & backbones for different methods.
Remaining issues for fair comparison.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>ImageNet</th>
<th>CUB</th>
<th>OpenImages</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VGG</td>
<td>Inception</td>
<td>ResNet</td>
</tr>
<tr>
<td>CAM '16</td>
<td>60.0</td>
<td>63.4</td>
<td>63.7</td>
</tr>
<tr>
<td>HaS '17</td>
<td>60.6</td>
<td>63.7</td>
<td>63.4</td>
</tr>
<tr>
<td>ACoL '18</td>
<td>57.4</td>
<td>63.7</td>
<td>62.3</td>
</tr>
<tr>
<td>SPG '18</td>
<td>59.9</td>
<td>63.3</td>
<td>63.3</td>
</tr>
<tr>
<td>ADL '19</td>
<td>59.9</td>
<td>61.4</td>
<td>63.7</td>
</tr>
<tr>
<td>CutMix '19</td>
<td>59.5</td>
<td>63.9</td>
<td>63.3</td>
</tr>
</tbody>
</table>

- Full 54 numbers = 6 methods x 3 datasets x 3 backbones.
That finalizes our benchmark contribution!

https://github.com/clovaai/wsolevaluation/
How do the previous WSOL methods compare?
Previous WSOL methods under the new benchmark

<table>
<thead>
<tr>
<th>Datasets</th>
<th>ImageNet</th>
<th></th>
<th></th>
<th>OpenImages</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VGG</td>
<td>Inception</td>
<td>ResNet</td>
<td>VGG</td>
<td>Inception</td>
<td>ResNet</td>
</tr>
<tr>
<td><strong>Backbone</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAM ‘16</td>
<td>60.0</td>
<td>63.4</td>
<td>63.7</td>
<td>63.7</td>
<td>56.7</td>
<td>63.0</td>
</tr>
<tr>
<td>HaS ‘17</td>
<td>60.6</td>
<td>63.7</td>
<td>63.4</td>
<td>63.7</td>
<td>53.4</td>
<td>64.6</td>
</tr>
<tr>
<td>ACoL ‘18</td>
<td>57.4</td>
<td>63.7</td>
<td>62.3</td>
<td>57.4</td>
<td>56.2</td>
<td>66.4</td>
</tr>
<tr>
<td>SPG ‘18</td>
<td>59.9</td>
<td>63.3</td>
<td>63.3</td>
<td>56.3</td>
<td>55.9</td>
<td>60.4</td>
</tr>
<tr>
<td>ADL ‘19</td>
<td>59.9</td>
<td>61.4</td>
<td>63.7</td>
<td>66.3</td>
<td>58.8</td>
<td>58.3</td>
</tr>
<tr>
<td>CutMix ‘19</td>
<td>59.5</td>
<td>63.9</td>
<td>63.3</td>
<td>62.3</td>
<td>57.4</td>
<td>62.8</td>
</tr>
</tbody>
</table>

- Is there a clear winner against the CAM in 2016?
What if the validation samples are used for model training?
Few-shot learning baseline.

- # Validation samples: 1-5 samples/class.
- What if they are used for training the model itself?
Few-shot learning results.

- FSL > WSOL at only 2-3 full supervision / class.
- FSL is an important baseline to compare against.
- New research directions: semi-weak supervision.
Takeaways

• "Weak supervision" may not really be a weak supervision.

• We propose a new evaluation protocol for WSOL task.

• Under the new protocol, there was no significant progress in WSOL methods.
Thank you