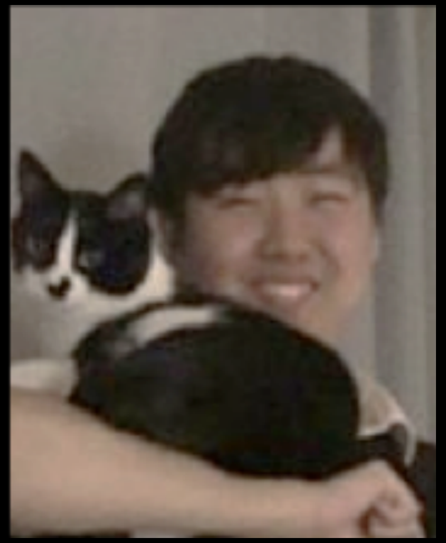
# Evaluating Weakly-Supervised Object Localization Methods Right

**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

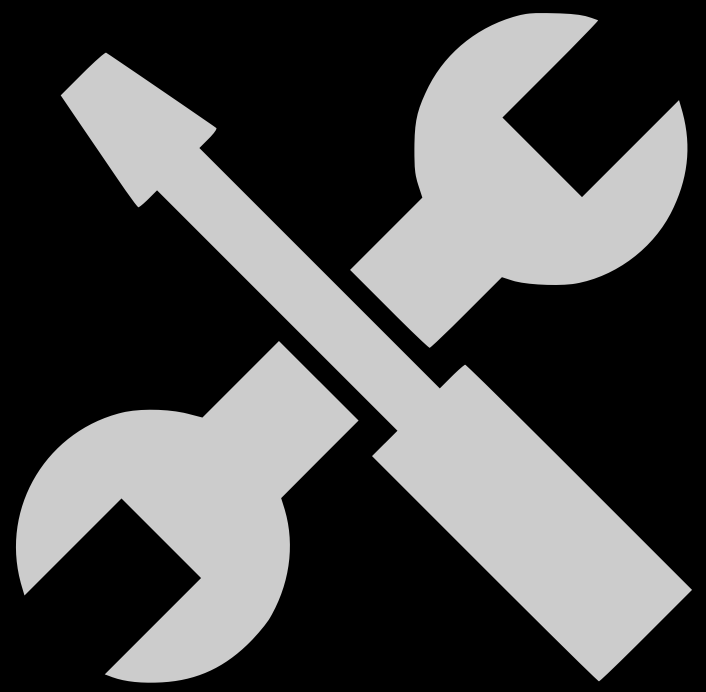
*** Equal contribution**

NAVER CLOVA

YONSEI UNIVERSITY 1885 연세대학교

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

# 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


Semantic segmentation


Object localization


Instance segmentation


Classification


Semantic segmentation


Object localization


Instance segmentation


Classification


Semantic segmentation


**Object localization**

Classify pixels by instance:

- The image **must** contain a single class.
- The class is known.
- FG-BG mask as final output.

Instance segmentation

Task goal: FG-BG mask

Task goal: FG-BG mask

## Supervision types


Weak supervision:
Class label

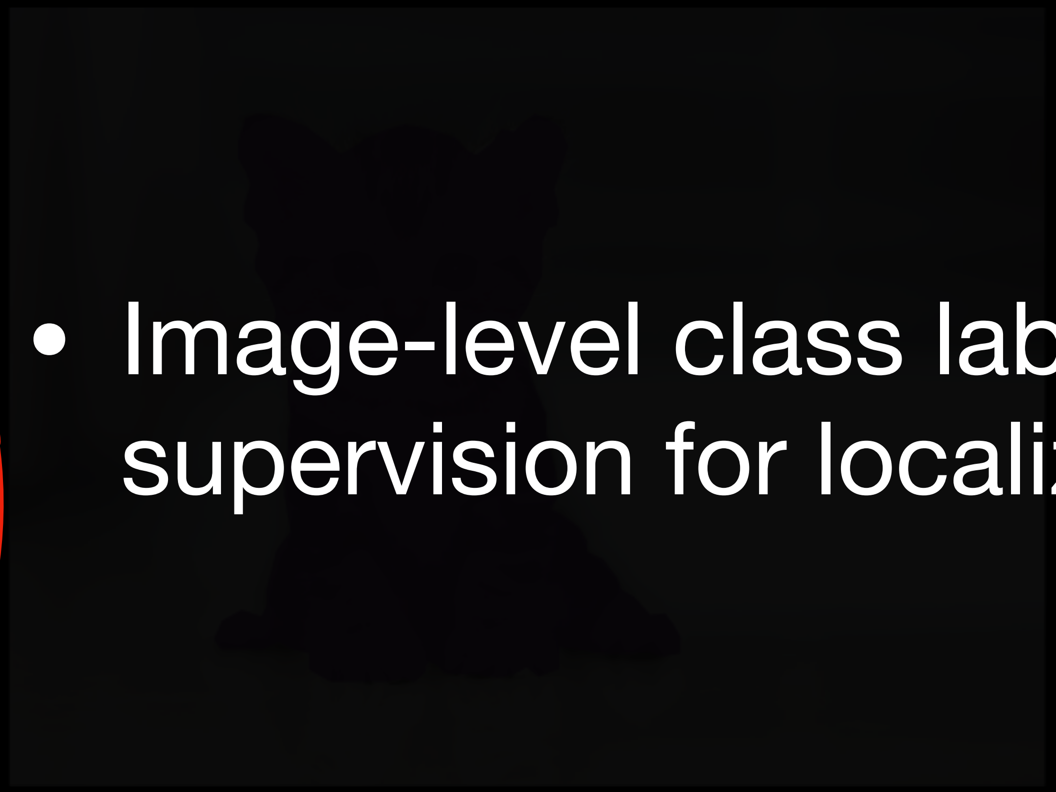
Full supervision:
FG-BG mask

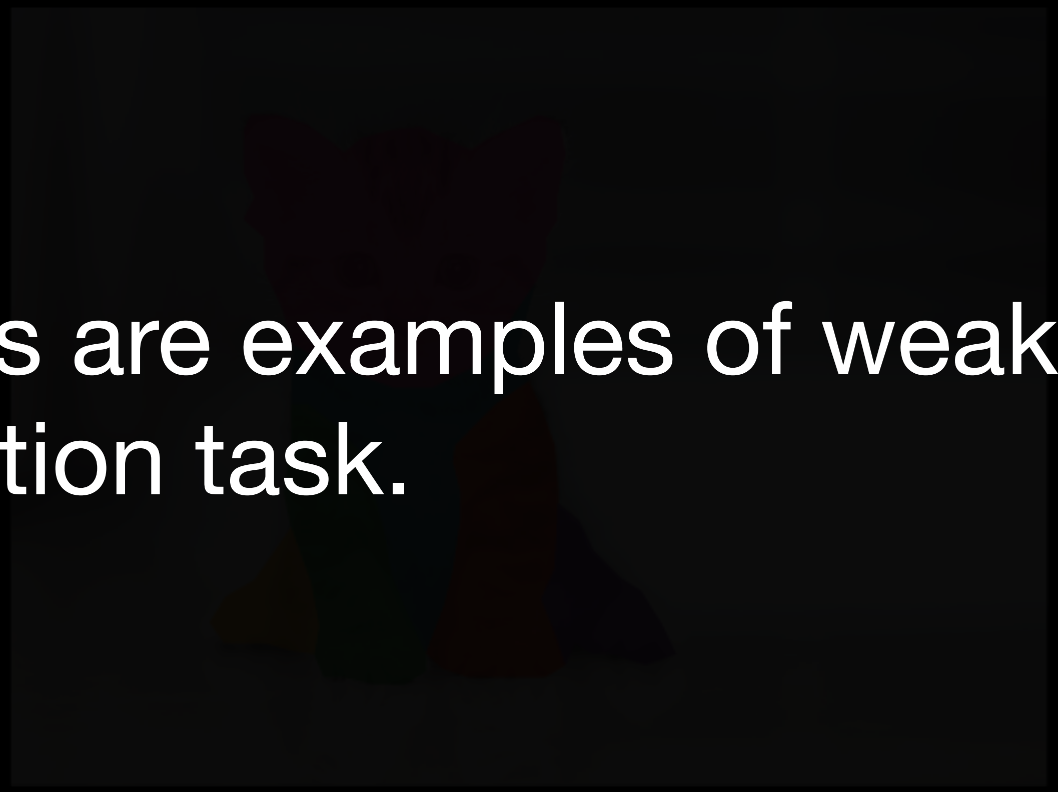
Strong supervision:
Part parsing mask

Task goal: FG-BG mask

## Supervision types


Weak supervision:
Class label





- Image-level class labels are examples of weak supervision for localization task.

Full supervision:
FG-BG mask

Strong supervision:
Part parsing mask

# Weakly-supervised object localization

**Test-time task: Localization.**

Input image

FG-BG mask

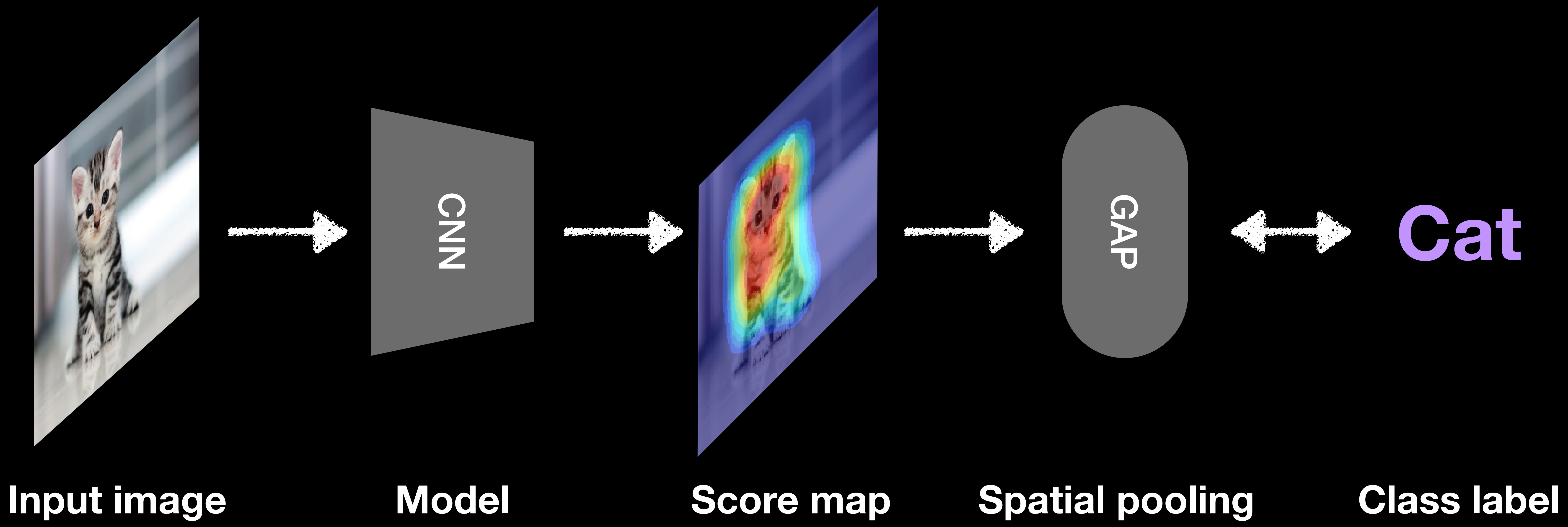
**Train-time supervision: Images + class labels**

Input image

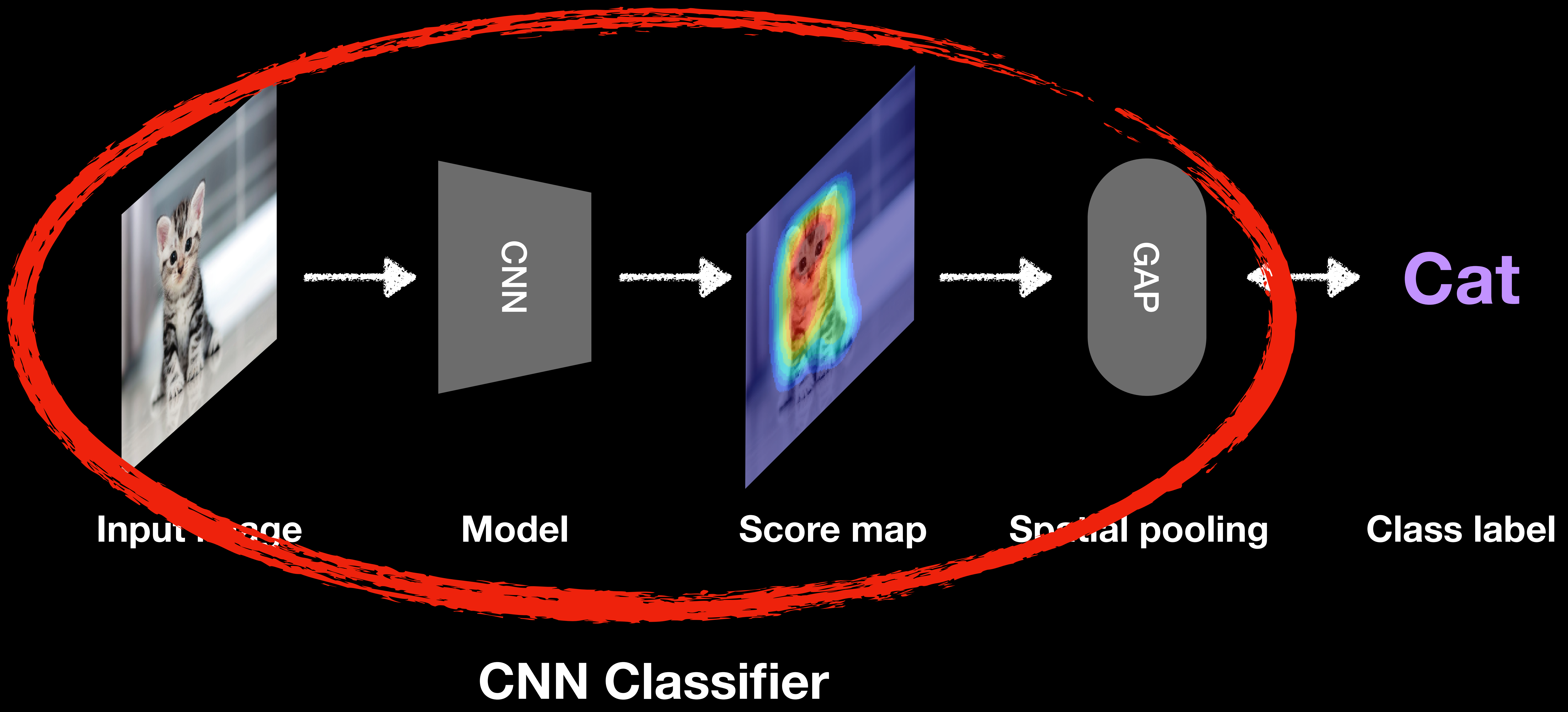
+

Cat

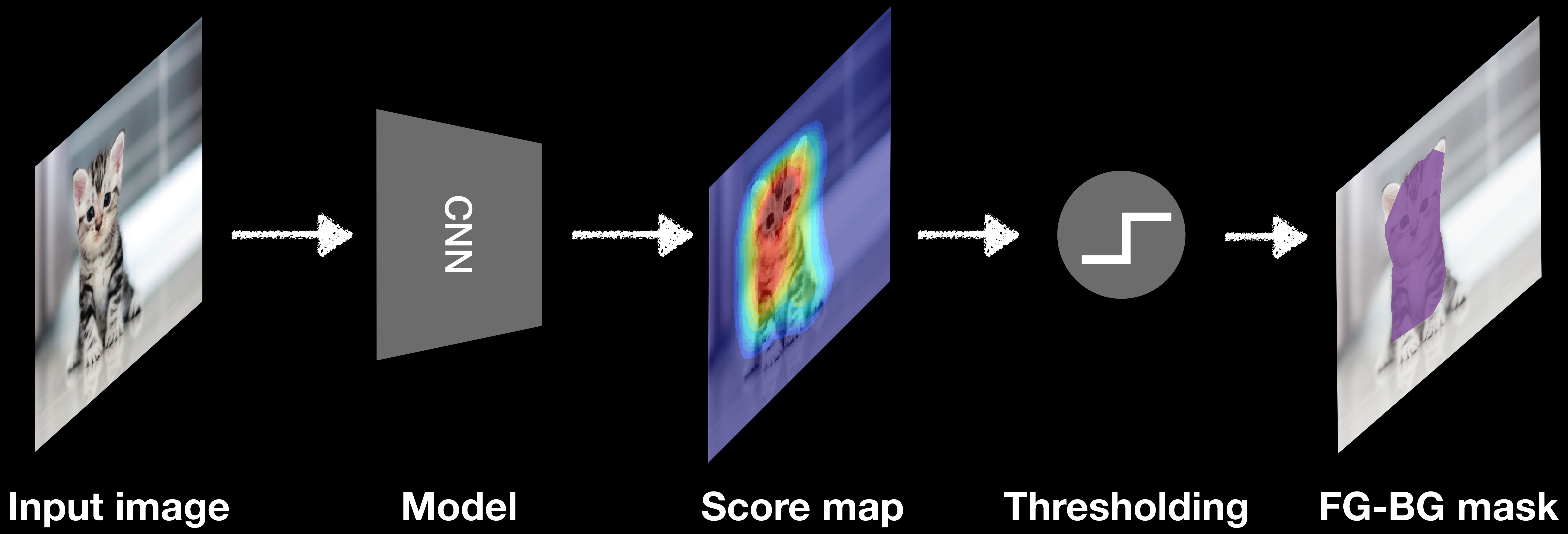
# How to train a WSOL model.
# CAM example (CVPR'16)



# How to train a WSOL model.
# CAM example (CVPR'16)

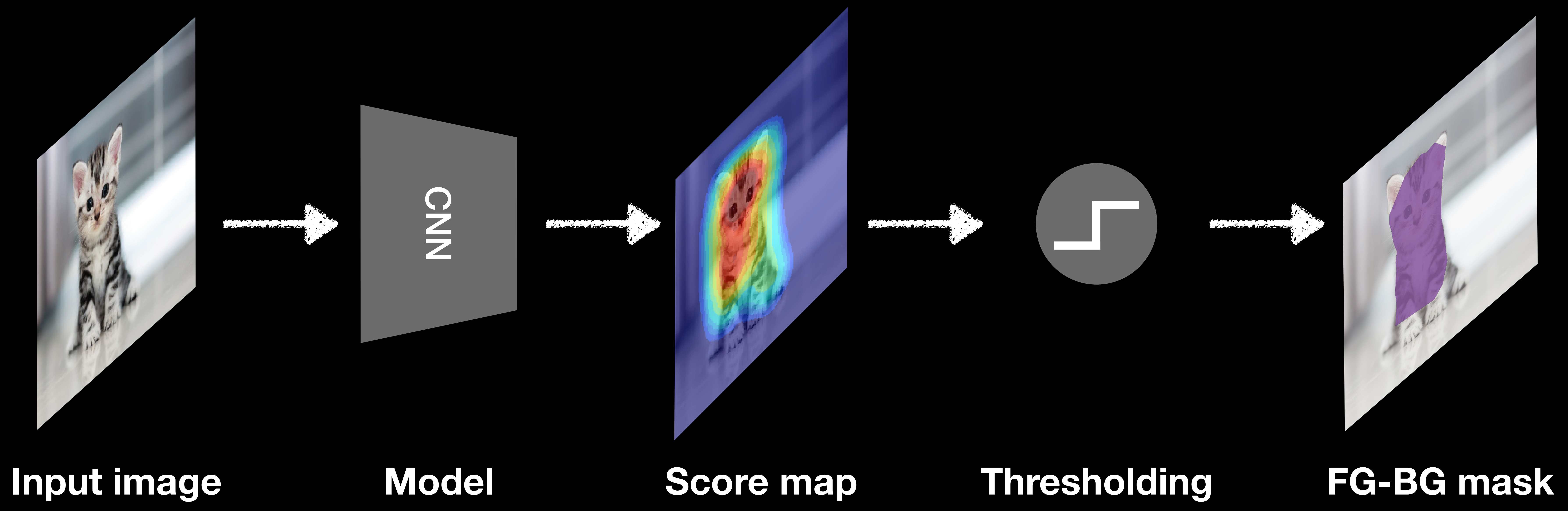


# CAM at test time.



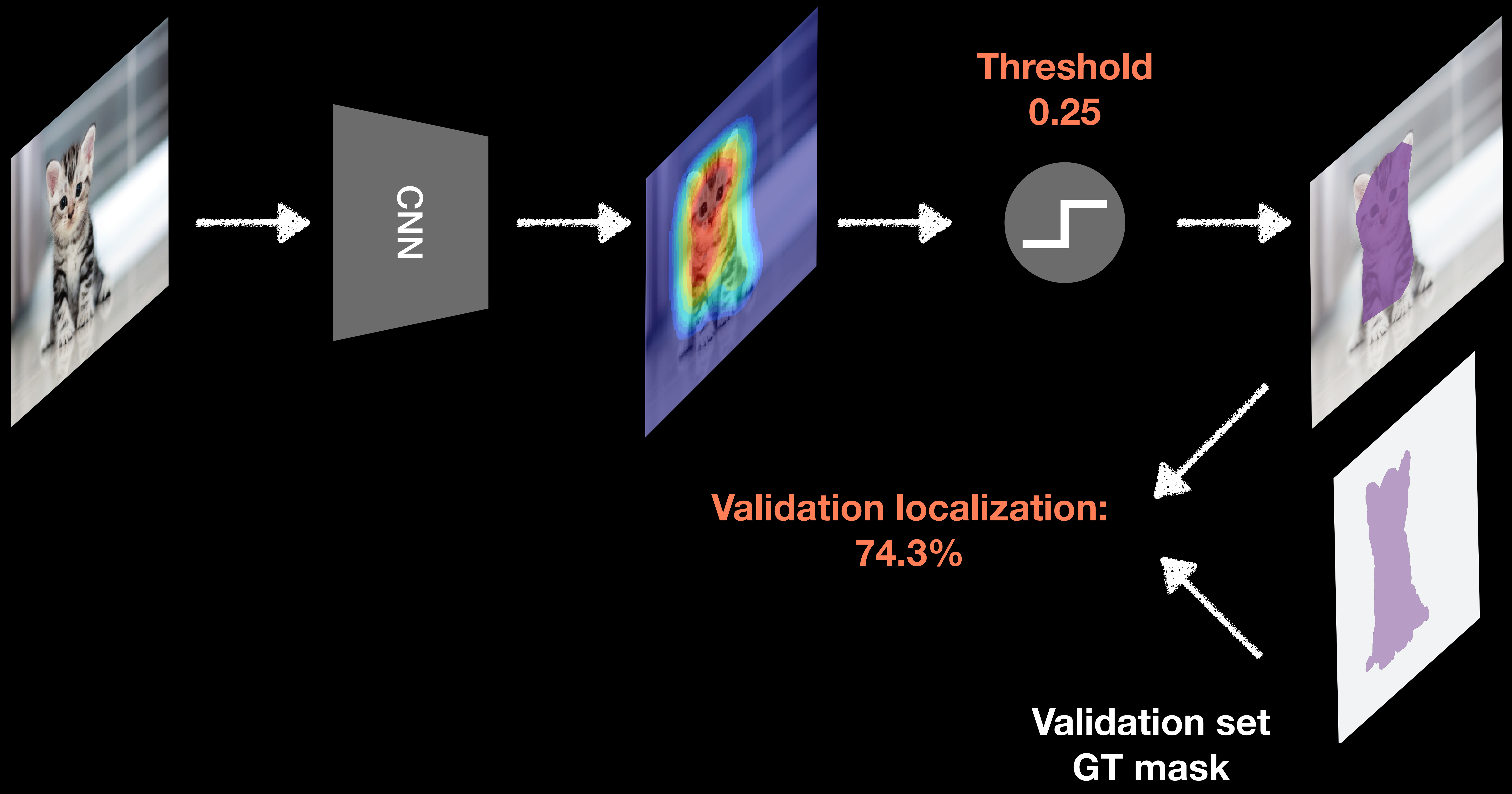
We didn't used any full supervision, did we?

# Implicit full supervision for WSOL.

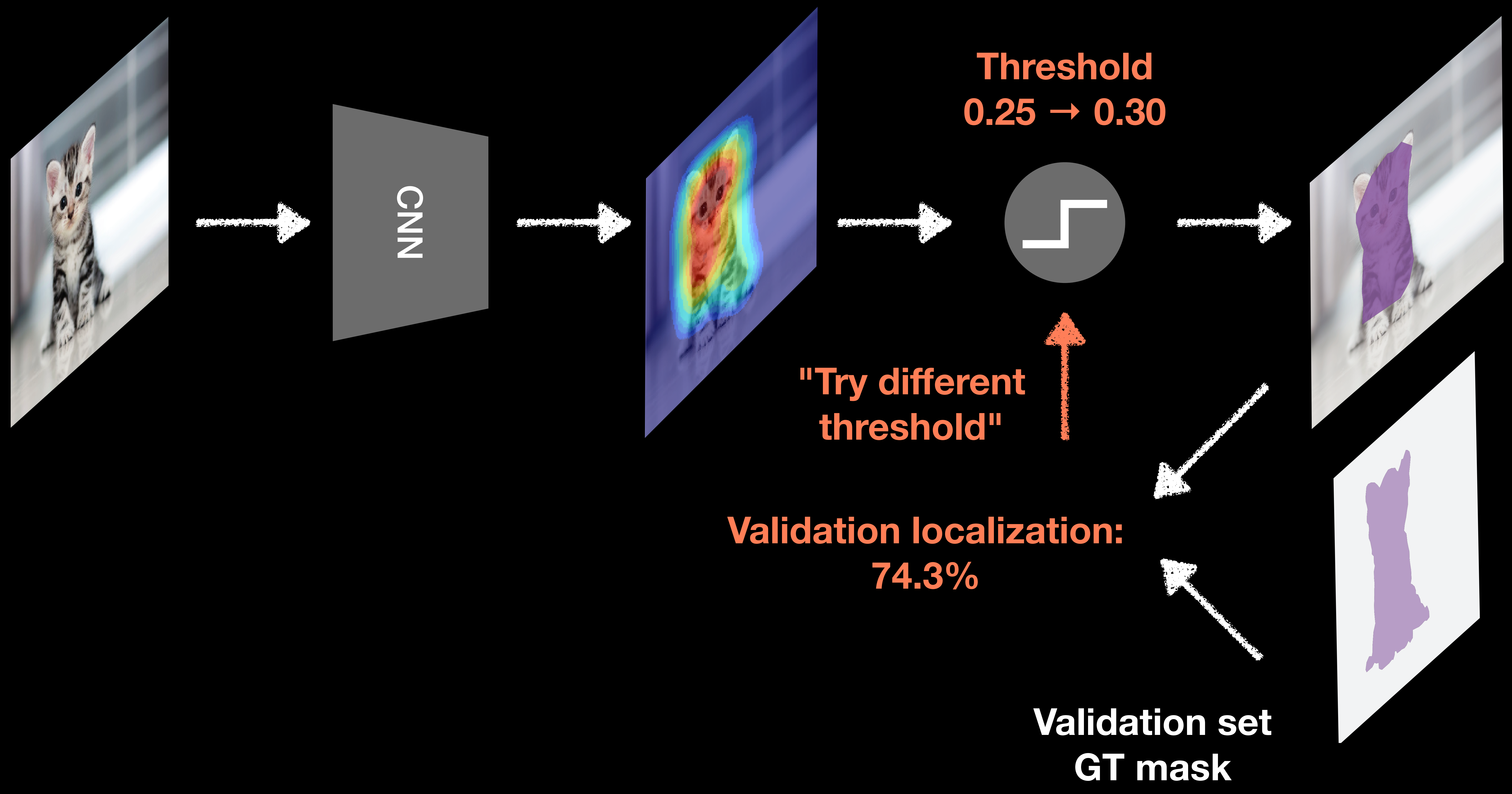



**Which threshold do we choose?**

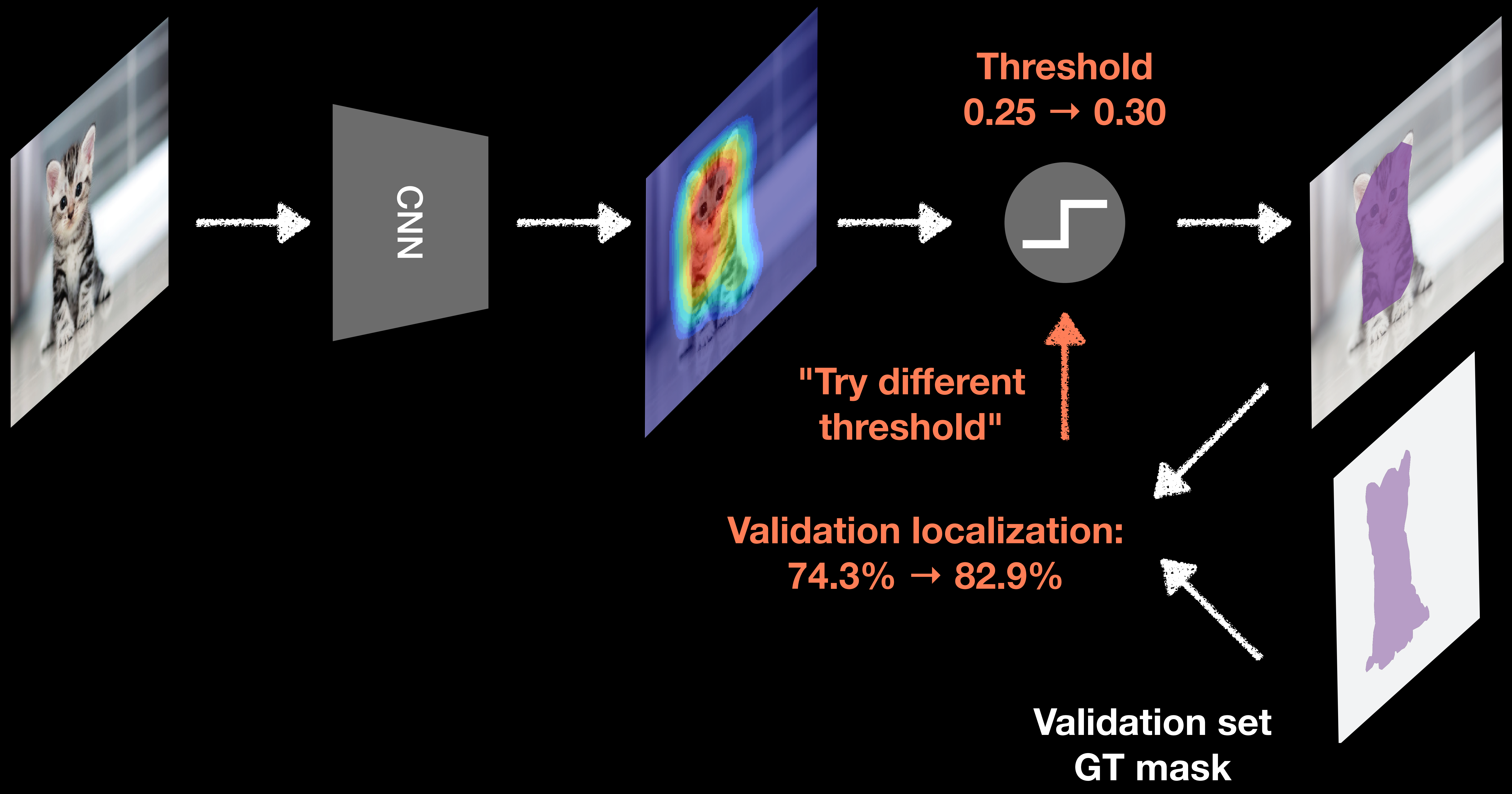
# Implicit full supervision for WSOL.



# Implicit full supervision for WSOL.



# Implicit full supervision for WSOL.



# WSOL methods have many hyperparameters to tune.

| Method | Hyperparameters |
| --- | --- |
| **CAM, CVPR'16** | Threshold / Learning rate / Feature map size |
| **HaS, ICCV'17** | Threshold / Learning rate / Feature map size / Drop rate / Drop area |
| **ACoL, CVPR'18** | Threshold / Learning rate / Feature map size / Erasing threshold |
| **SPG, ECCV'18** | Threshold / Learning rate / Feature map size / Threshold 1L / Threshold 1U / Threshold 2L / Threshold 2U / Threshold 3L / Threshold 3U |
| **ADL, CVPR'19** | Threshold / Learning rate / Feature map size / Drop rate / Erasing threshold |
| **CutMix, ICCV'19** | Threshold / Learning rate / Feature map size / Size prior / Mix rate |

- 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.

| Dataset | Training set (Weak sup) | Validation set (Full sup) | Test set (Full sup) |
|---|---|---|---|
| ImageNet | ✔ | ✘ ImageNetV2[a] exists, but no full sup. | ✔ |
| CUB | ✔ | ✘ No images, nothing. | ✔ |

[a] Recht et al. Do ImageNet classifiers generalize to ImageNet? ICML 2019.

# Our benchmark proposal.

| Dataset | Training set (Weak sup) | Validation set (Full sup) | Test set (Full sup) |
|---|---|---|---|
| **ImageNet** | ✓ | ✓ ImageNetV2 + Our annotations. | ✓ |
| **CUB** | ✓ | ✓ Our image collections + Our annotations. | ✓ |
| **OpenImages** | ✓ Curation of OpenImages30k train set. | ✓ Curation of OpenImages30k val set. | ✓ Curation of OpenImages30k test set. |

# Our benchmark proposal.

| Dataset | Training set (Weak sup) | Validation set (Full sup) | Test set (Full sup) |
|---|---|---|---|
| ImageNet | ✓ | ✓ ImageNetV2 + Our annotations. | ✓ |
| CUB | ✓ | ✓ Our image collections + Our annotations. | ✓ |
| OpenImages | ✓ Curation of OpenImages30k train set. | ✓ Curation of OpenImages30k val set. | ✓ Curation of OpenImages30k test set. |

Newly introduced dataset.

# Do the validation explicitly, with the *same* search algorithm.

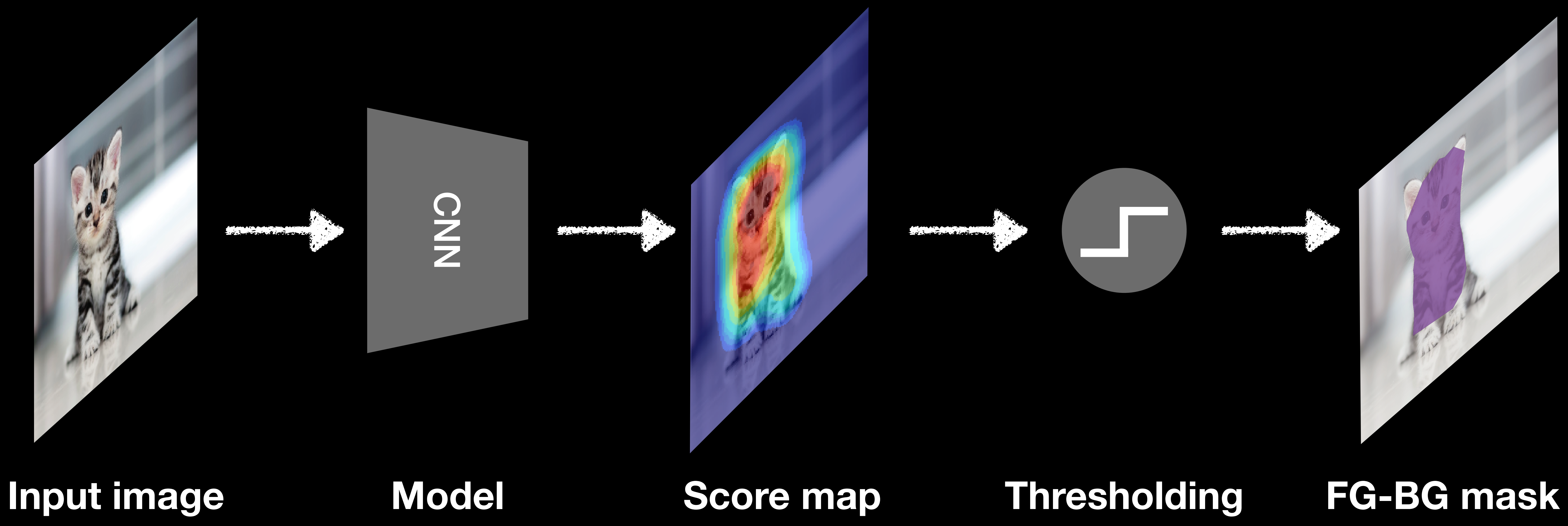
For each WSOL method, tune hyperparameters with

- Optimization algorithm: Random search.
- Search space: Feasible range (not "reasonable range").
- Search iteration: 30 tries.

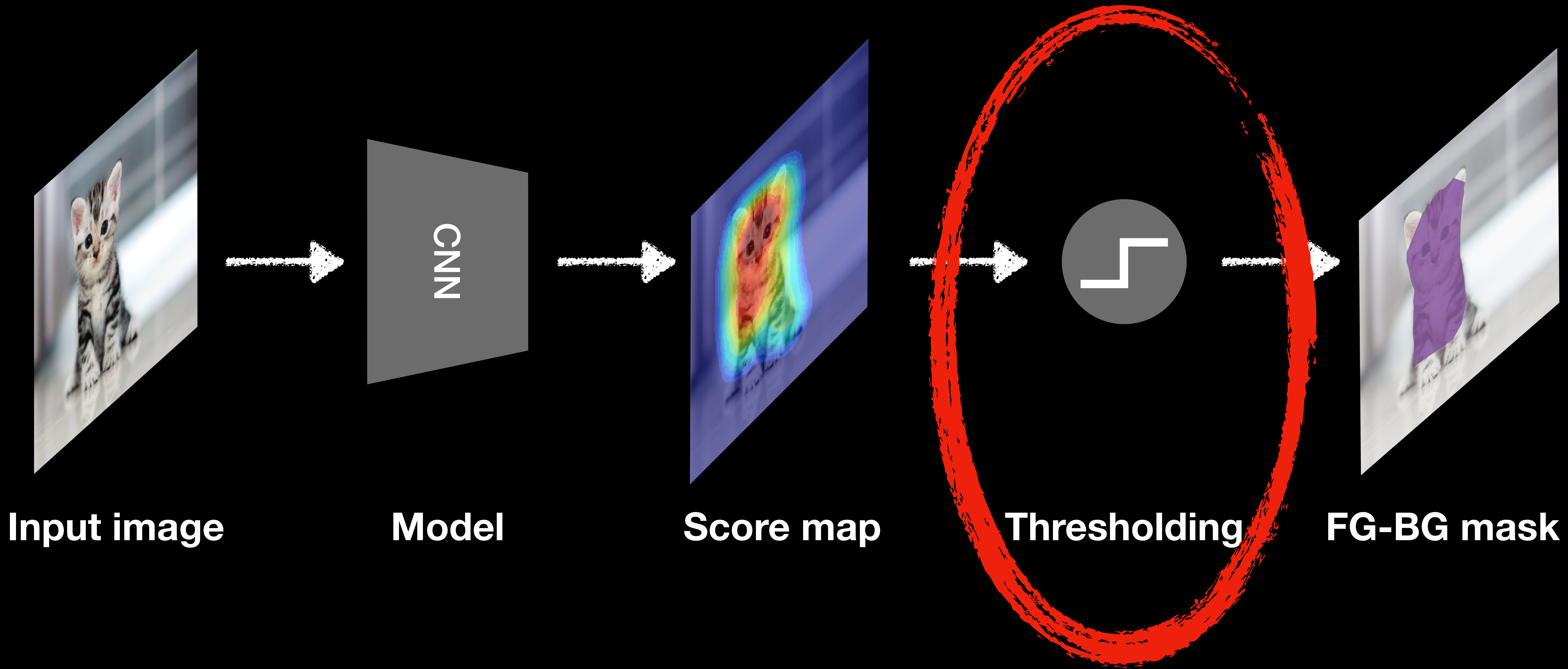
# Do the validation explicitly, with the *same* search algorithm.

| Method | Hyperparameters | Search space (Feasible range) |
|---|---|---|
| **CAM, CVPR'16** | Learning rate<br>Feature map size | LogUniform[0.00001,1]<br>Categorical{14,28} |
| **HaS, ICCV'17** | Learning rate<br>Feature map size<br>Drop rate<br>Drop area | LogUniform[0.00001,1]<br>Categorical{14,28}<br>Uniform[0,1]<br>Uniform[0,1] |
| **ACoL, CVPR'18** | Learning rate<br>Feature map size<br>Erasing threshold | LogUniform[0.00001,1]<br>Categorical{14,28}<br>Uniform[0,1] |
| **SPG, ECCV'18** | Learning rate<br>Feature map size<br>Threshold 1L<br>Threshold 1U<br>Threshold 2L<br>Threshold 2U | LogUniform[0.00001,1]<br>Categorical{14,28}<br>Uniform[0,d1]<br>Uniform[d1,1]<br>Uniform[0,d2]<br>Uniform[d2,1] |
| **ADL, CVPR'19** | Learning rate<br>Feature map size<br>Drop rate<br>Erasing threshold | LogUniform[0.00001,1]<br>Categorical{14,28}<br>Uniform[0,1]<br>Uniform[0,1] |
| **CutMix, ICCV'19** | Learning rate<br>Feature map size<br>Size prior<br>Mix rate | LogUniform[0.00001,1]<br>Categorical{14,28}<br>1/Uniform(0,2]-1/2<br>Uniform[0,1] |

# Previous treatment of the score map threshold.



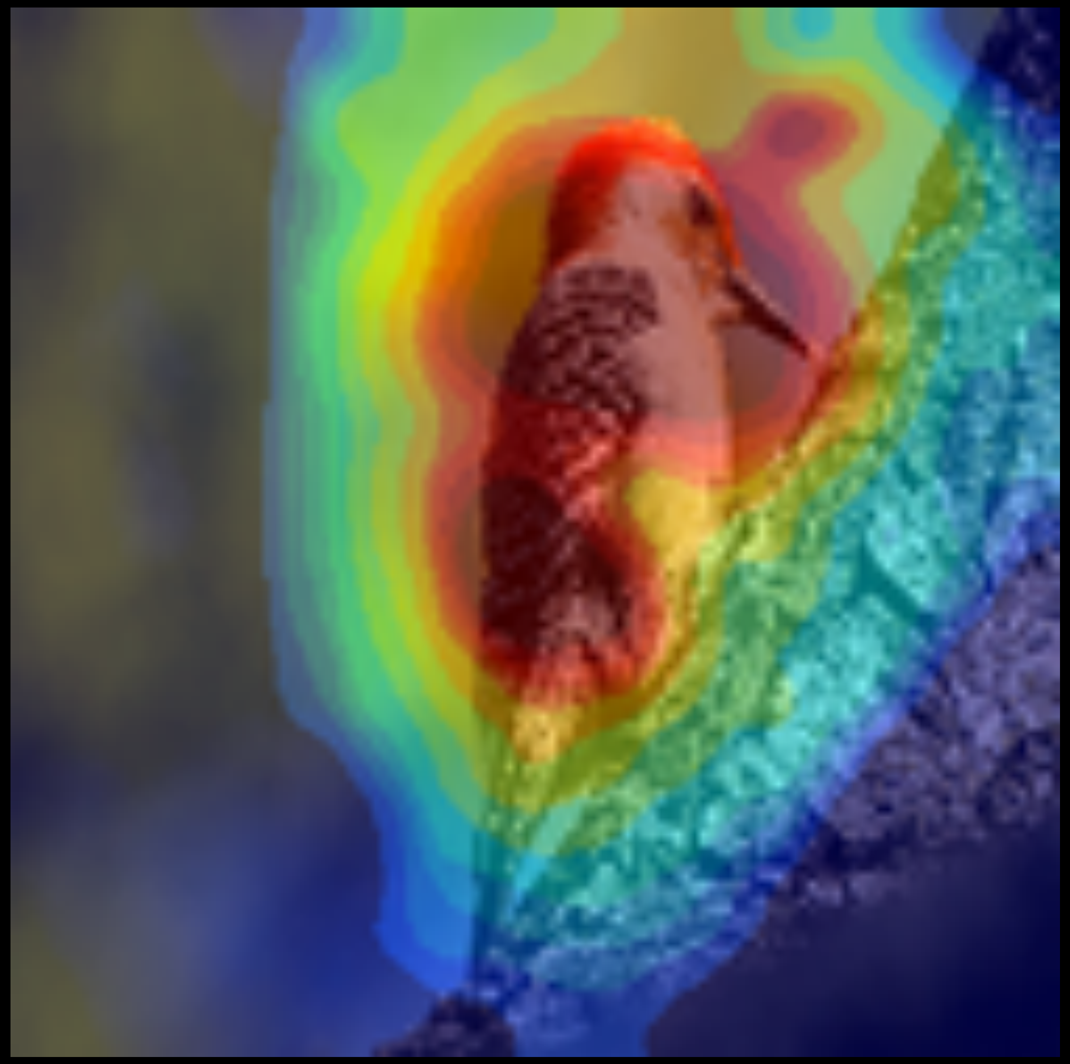
# Previous treatment of the score map threshold.



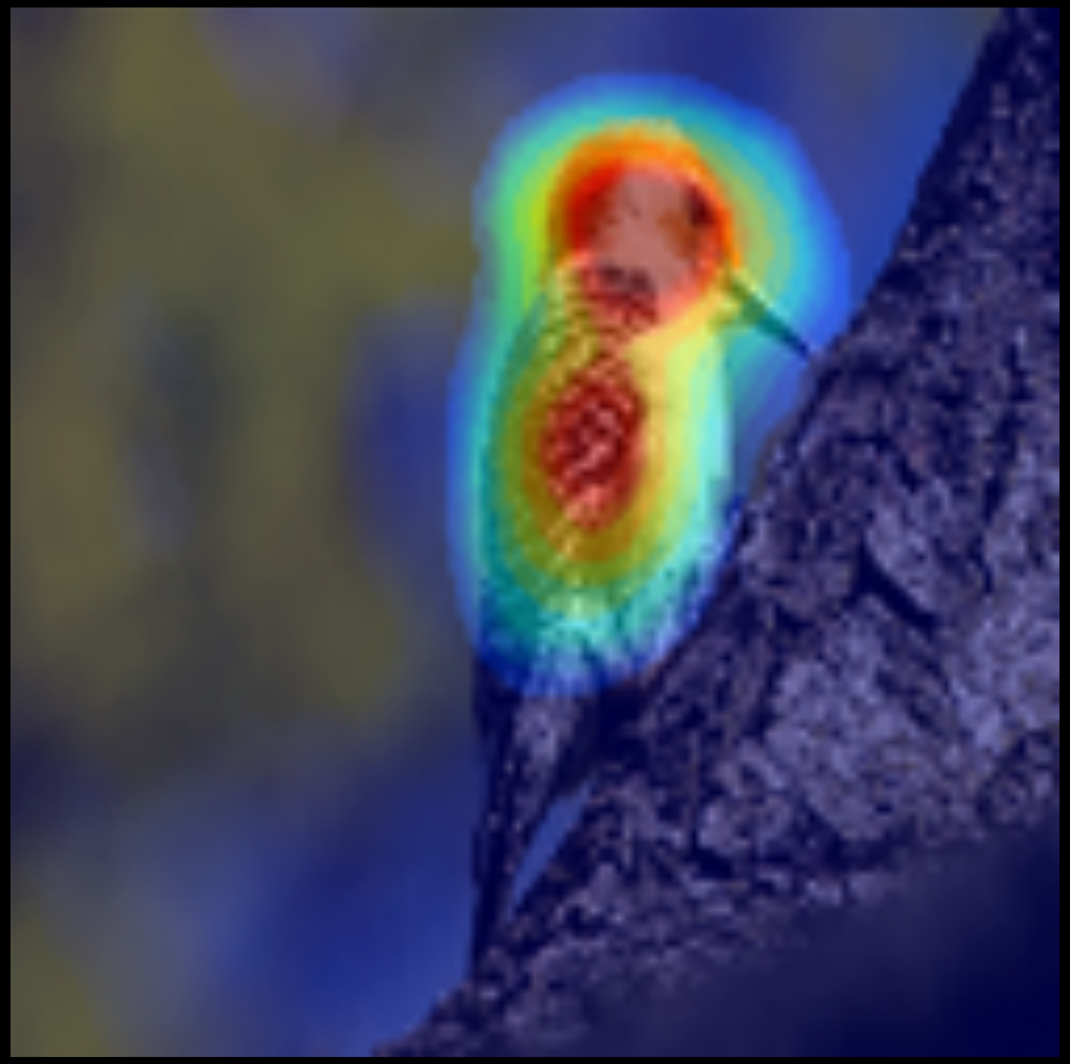

- Score maps are natural outputs of WSOL methods.
- The binarizing threshold is sometimes tuned, sometimes set as a "common" value.

# But setting the right threshold is critical.

Input image



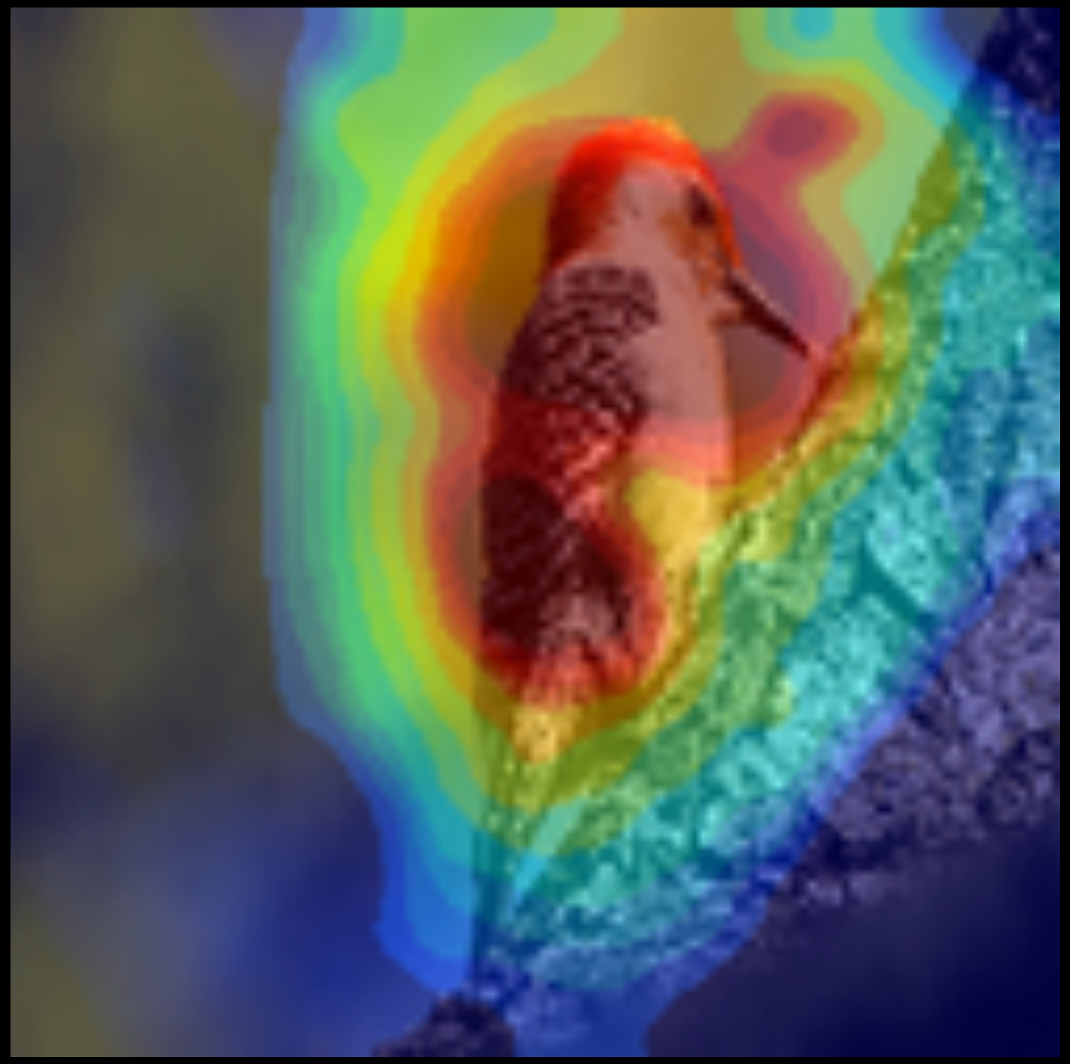
Score map of Method 1



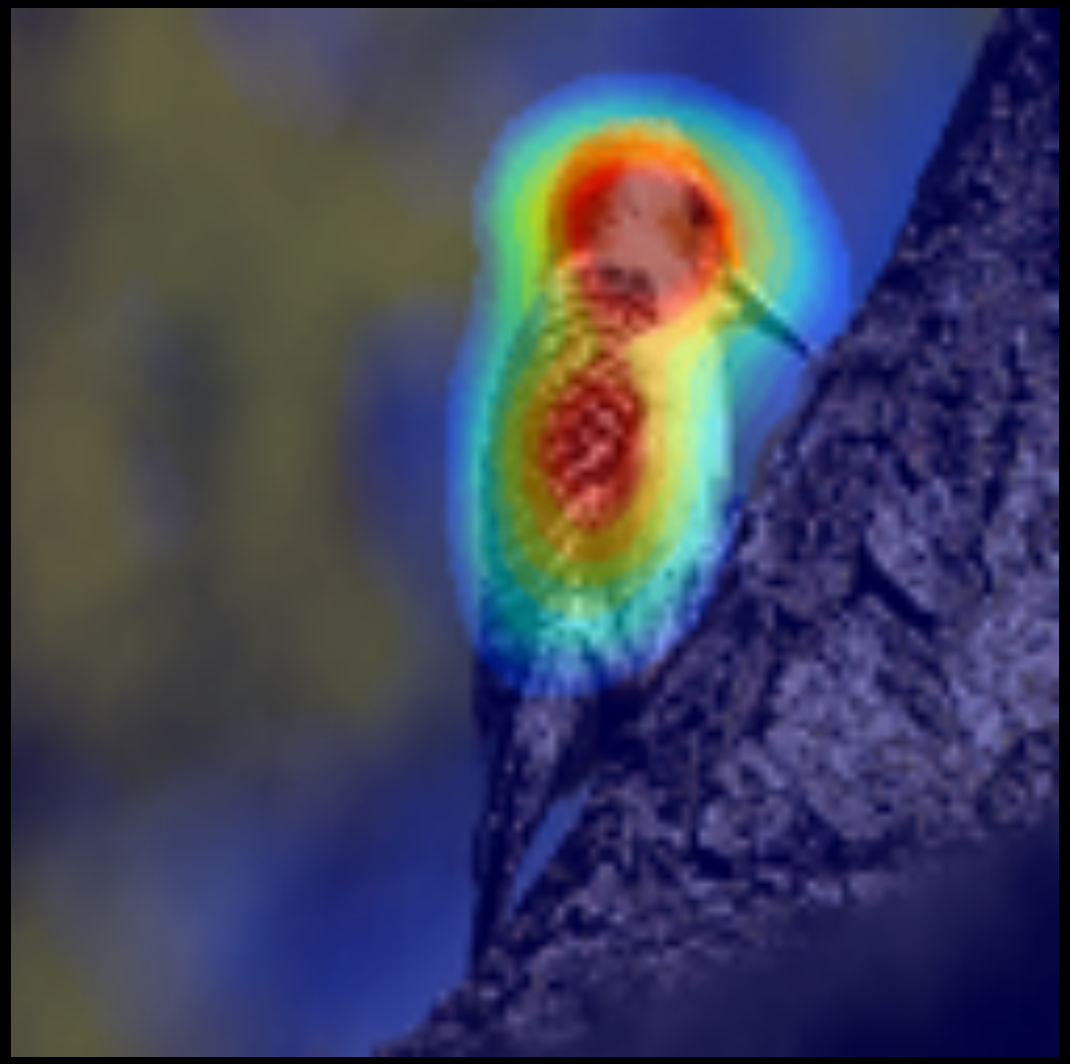
Score map of Method 2

# But setting the right threshold is critical.

Input image



Score map of Method 1

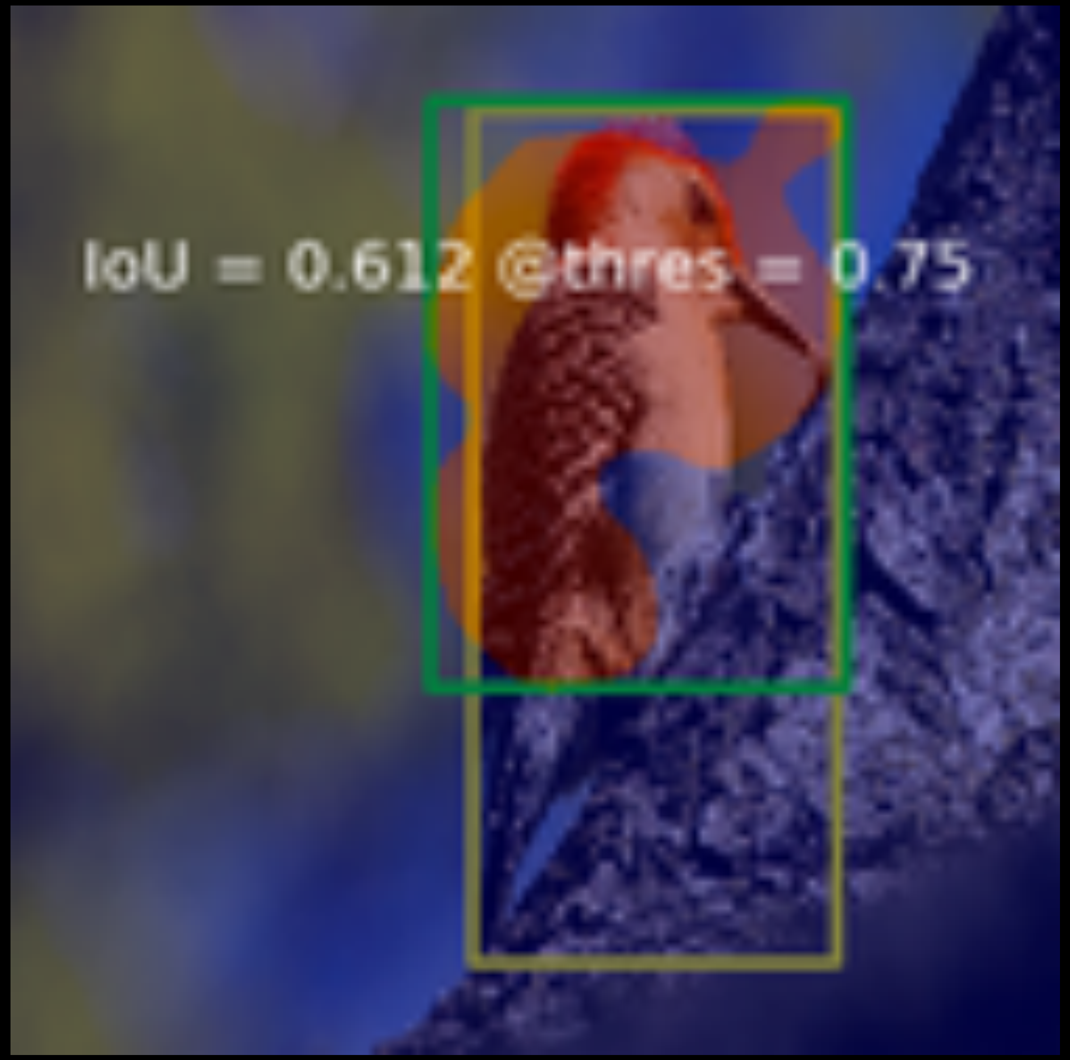


Score map of Method 2

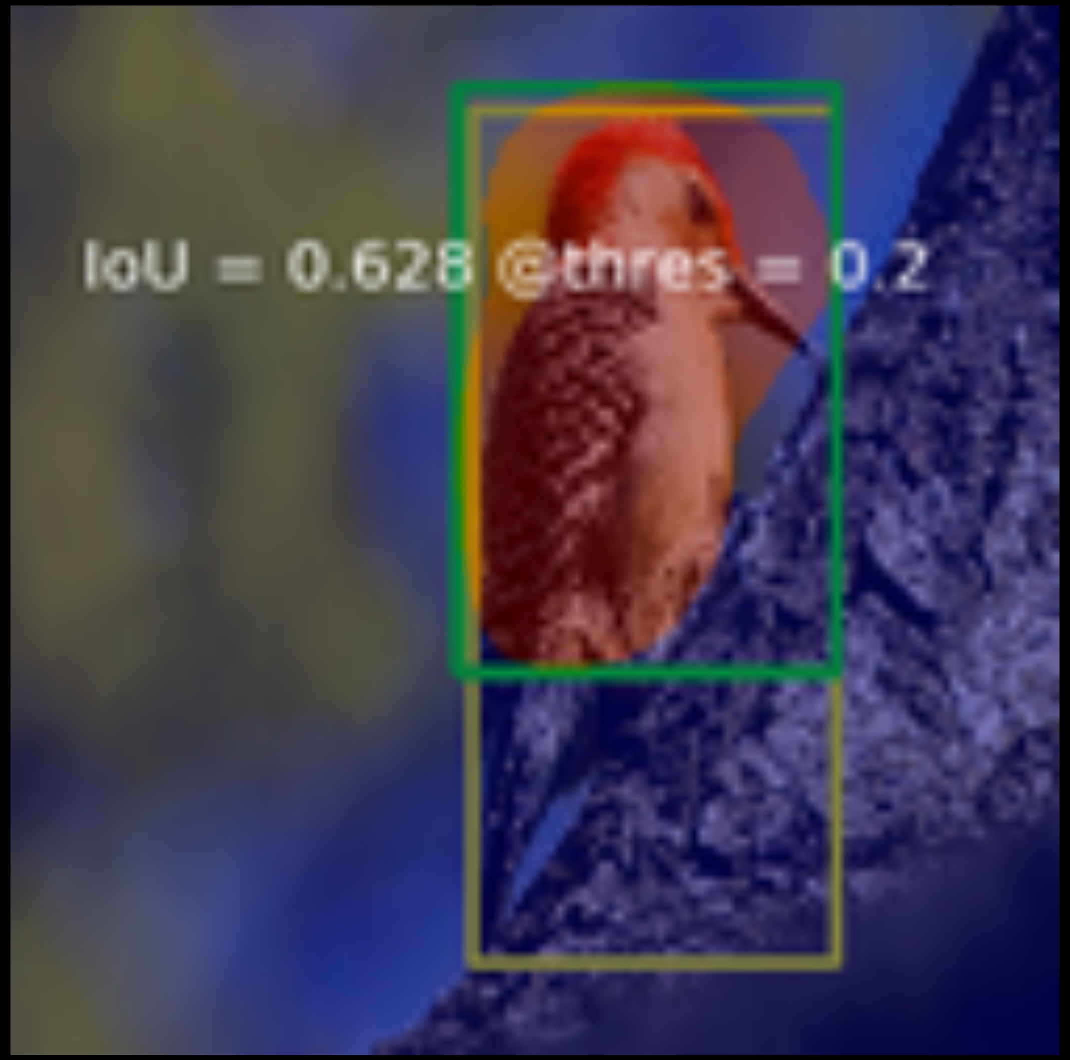
- Method 1 seems to perform better: it covers the object extent better.

# But setting the right threshold is critical.

Input image




Score map of Method 1




Score map of Method 2

- 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.

| Datasets | ImageNet | | | CUB | | |
|---|---|---|---|---|---|---|
| Backbone | VGG | Inception | ResNet | VGG | Inception | ResNet |
| CAM '16 | 42.8 | - | 46.3 | 37.1 | 43.7 | 49.4 |
| HaS '17 | - | - | - | - | - | - |
| ACoL '18 | 45.8 | - | - | 45.9 | - | - |
| SPG '18 | - | 48.6 | - | - | 46.6 | - |
| ADL '19 | 44.9 | 48.7 | - | 52.4 | 53.0 | - |
| CutMix '19 | 43.5 | - | 47.3 | - | 52.5 | 54.8 |

- Different datasets & backbones for different methods.

# Remaining issues for fair comparison.

| Datasets | ImageNet | | | CUB | | | OpenImages | | |
|---|---|---|---|---|---|---|---|---|---|
| **Backbone** | VGG | Inception | ResNet | VGG | Inception | ResNet | VGG | Inception | ResNet |
| **CAM '16** | 60.0 | 63.4 | 63.7 | 63.7 | 56.7 | 63.0 | 58.3 | 63.2 | 58.5 |
| **HaS '17** | 60.6 | 63.7 | 63.4 | 63.7 | 53.4 | 64.6 | 58.1 | 58.1 | 55.9 |
| **ACoL '18** | 57.4 | 63.7 | 62.3 | 57.4 | 56.2 | 66.4 | 54.3 | 57.2 | 57.3 |
| **SPG '18** | 59.9 | 63.3 | 63.3 | 56.3 | 55.9 | 60.4 | 58.3 | 62.3 | 56.7 |
| **ADL '19** | 59.9 | 61.4 | 63.7 | 66.3 | 58.8 | 58.3 | 58.7 | 56.9 | 55.2 |
| **CutMix '19** | 59.5 | 63.9 | 63.3 | 62.3 | 57.4 | 62.8 | 58.1 | 62.6 | 57.7 |

- 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?

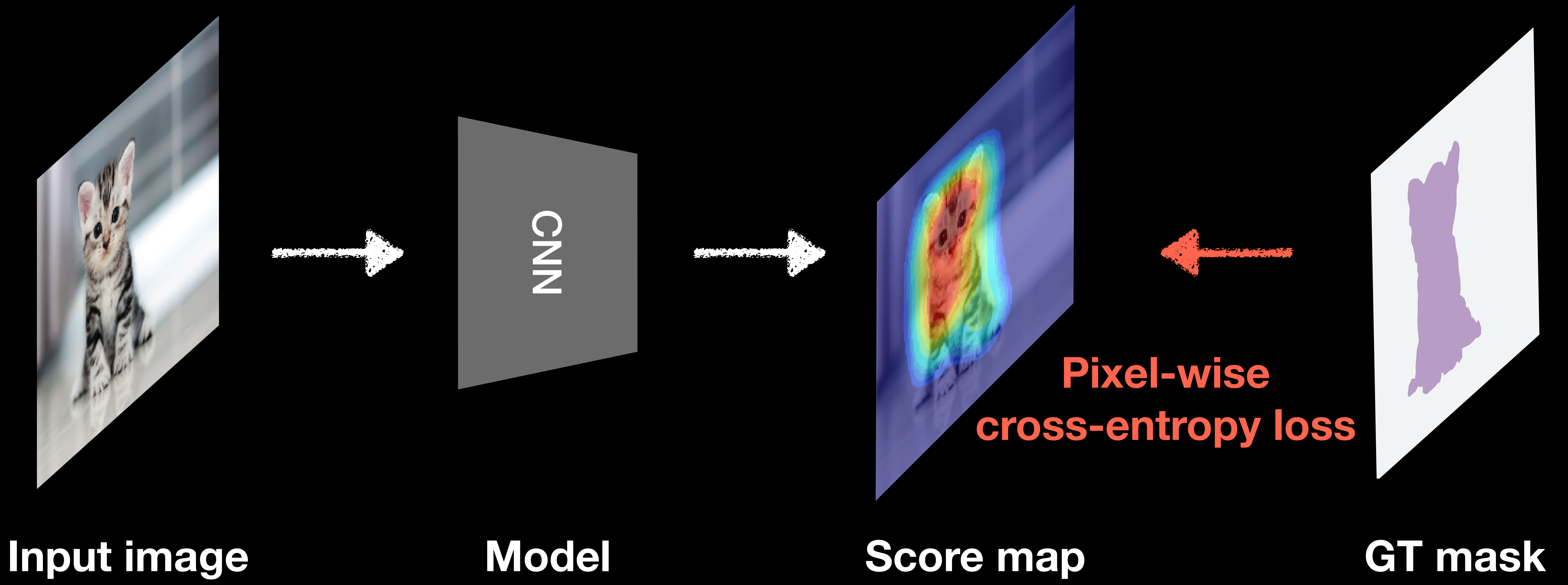
# Previous WSOL methods under the new benchmark

| Datasets | ImageNet | | | CUB | | | OpenImages | | |
|---|---|---|---|---|---|---|---|---|---|
| **Backbone** | VGG | Inception | ResNet | VGG | Inception | ResNet | VGG | Inception | ResNet |
| **CAM '16** | 60.0 | 63.4 | 63.7 | 63.7 | 56.7 | 63.0 | 58.3 | 63.2 | 58.5 |
| **HaS '17** | 60.6 | 63.7 | 63.4 | 63.7 | 53.4 | 64.6 | 58.1 | 58.1 | 55.9 |
| **ACoL '18** | 57.4 | 63.7 | 62.3 | 57.4 | 56.2 | 66.4 | 54.3 | 57.2 | 57.3 |
| **SPG '18** | 59.9 | 63.3 | 63.3 | 56.3 | 55.9 | 60.4 | 58.3 | 62.3 | 56.7 |
| **ADL '19** | 59.9 | 61.4 | 63.7 | 66.3 | 58.8 | 58.3 | 58.7 | 56.9 | 55.2 |
| **CutMix '19** | 59.5 | 63.9 | 63.3 | 62.3 | 57.4 | 62.8 | 58.1 | 62.6 | 57.7 |

- Is there a clear winner against the CAM in 2016?

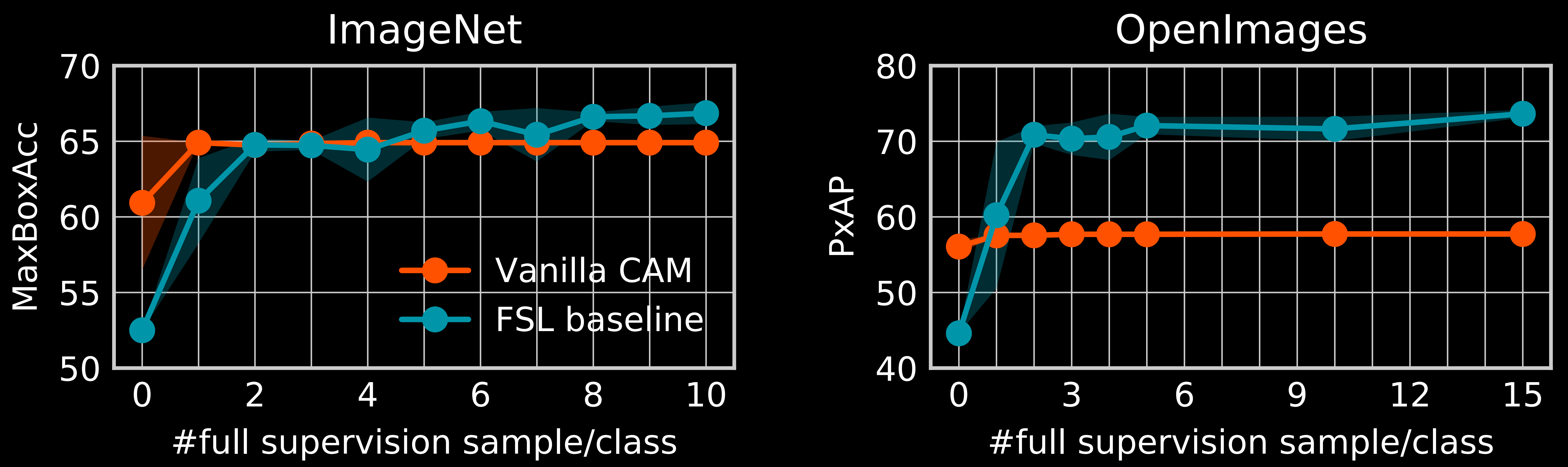
What if
the validation samples are
used for model training?

# 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