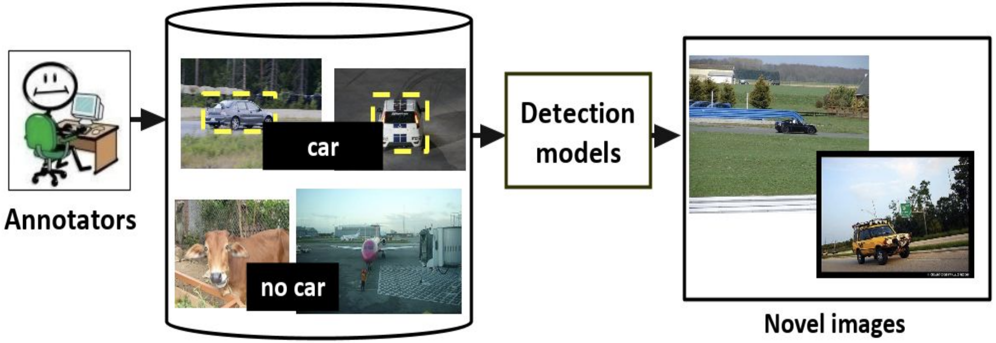
Hide-and-Seek: Forcing a Network to be Meticulous for Weakly-supervised Object and Action Localization

Krishna Kumar Singh and Yong Jae Lee
University of California, Davis

Introduction

Why not use the fully-supervised approach?



[Felzenszwalb et al. PAMI 2010, Girshick et al. CVPR 2014, Girshick ICCV 2015, ...]

Requires expensive, error-prone bounding box annotations.

Thus, it's not scalable

Weakly-supervised approach

- Visual classification and localization tasks

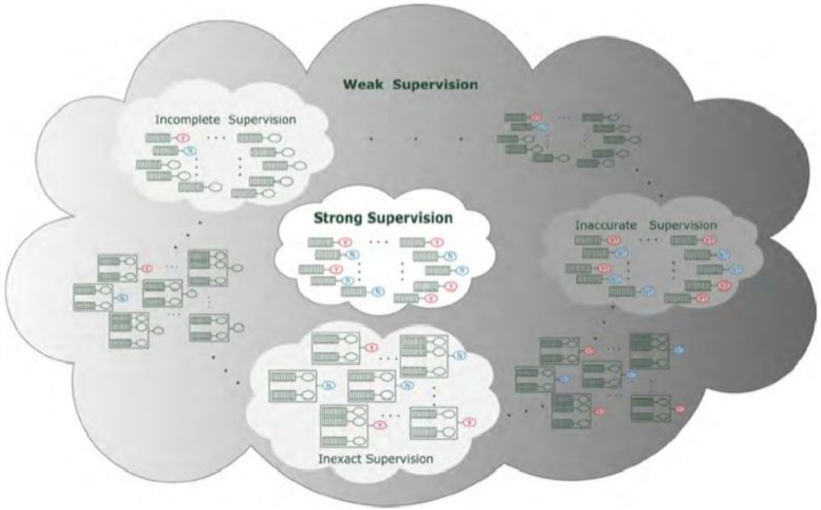
- Visual attribute localization

- **Requires less detailed annotations compared to the fully-supervised approach**

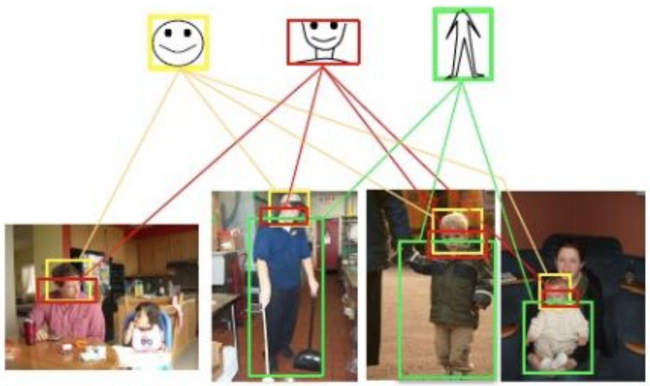
Weakly-supervised approach

- Supervision is provided at the image-level. It is scalable.

- Most weakly-supervised object localization approaches mine discriminative features or patches in the data that frequently appear in one class and rarely in other classes



Prior attempts to improve weak object localization



[Song et al. NIPS 2014]

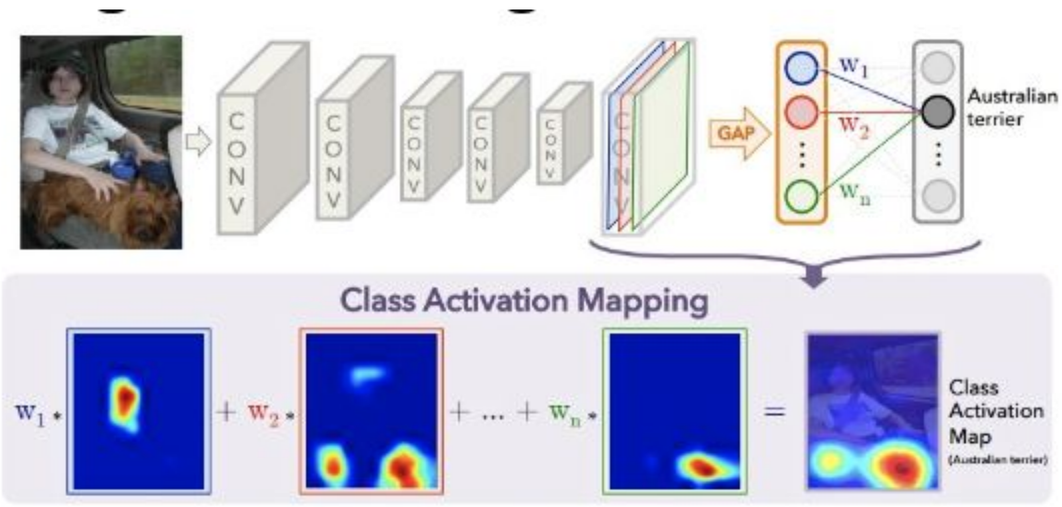
Select multiple discriminative regions

Prior attempts to improve weak object localization

[Singh et al. CVPR 2016]

Transfer tracked objects from videos to images

Prior attempts to improve weak object localization



[Zhou et al. CVPR 2016]

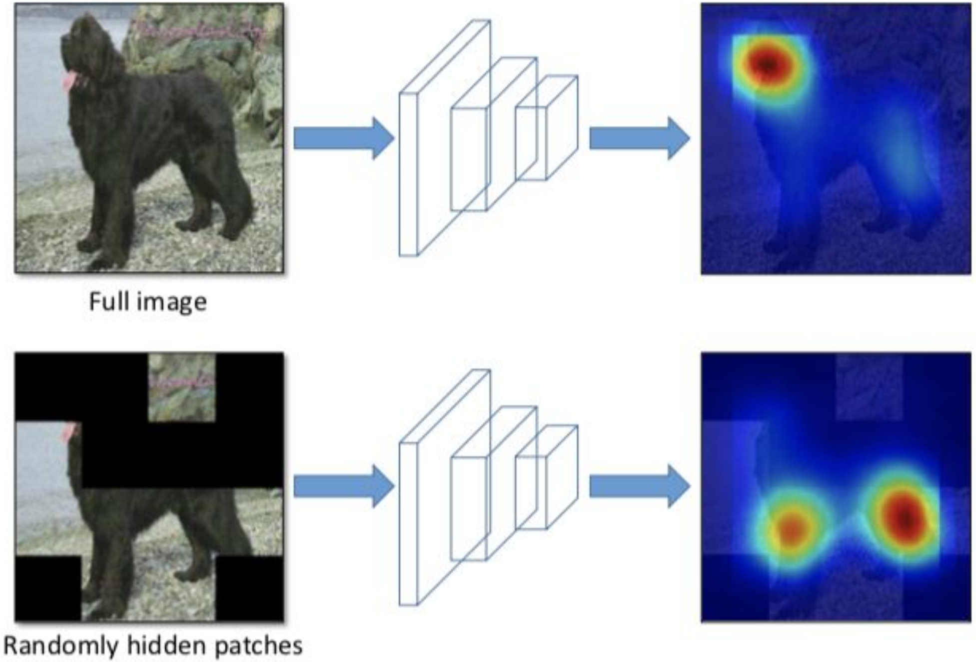
Global average pooling to encourage network to look at all relevant parts.

Hide-and-Seek

- If we randomly remove some patches from the image, the model must seek other relevant parts

- Hide-and-Seek only alters the input image

Hide-and-Seek



Related Work

- **Weakly-supervised object localization**
- Masking pixels or activations
- Action localization

Related Work

- **Weakly-supervised object localization**

  Other network architectures have been designed for weakly-supervised object detection, still rely on a classification objective and thus to fail capture the full extent of an object

- Masking pixels or activations

- Action localization

Related Work

- Weakly-supervised object localization

- **Masking pixels or activations**

  -In the paper, image regions are masked during training.

- Action localization

Related Work

- Weakly-supervised object localization
- Masking pixels or activations
- **Action localization**

Related Work

- Weakly-supervised object localization

- Masking pixels or activations

- **Action localization**

  -Fully-supervised methods/Weak-supervised methods/approach in the paper

Approach

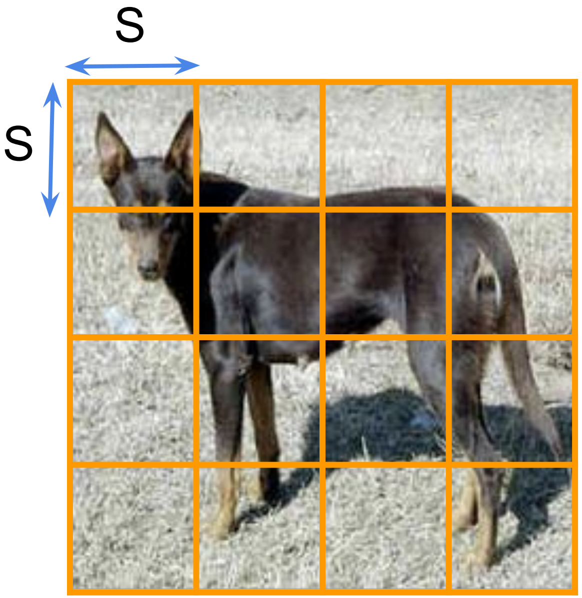
Hide-and-Seek (HaS) for:

- Weakly-supervised object localization in images
- Weakly-supervised temporal action localization in videos

Divide the training image into a grid with a patch size of S x S

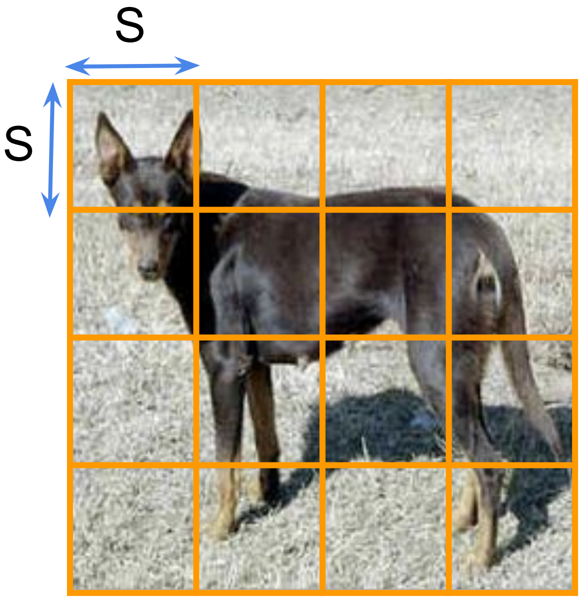
Training image
with label “dog”

Divide the training image into a grid with a patch size of S x S

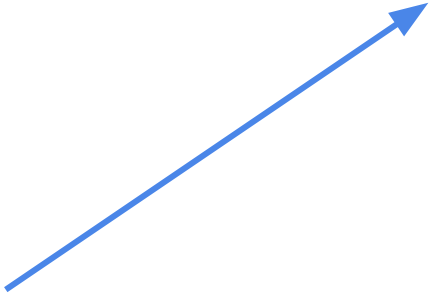


Training image
with label “dog”

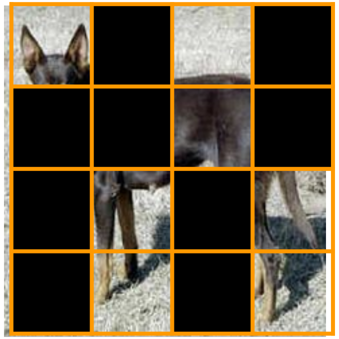
Randomly hide patches



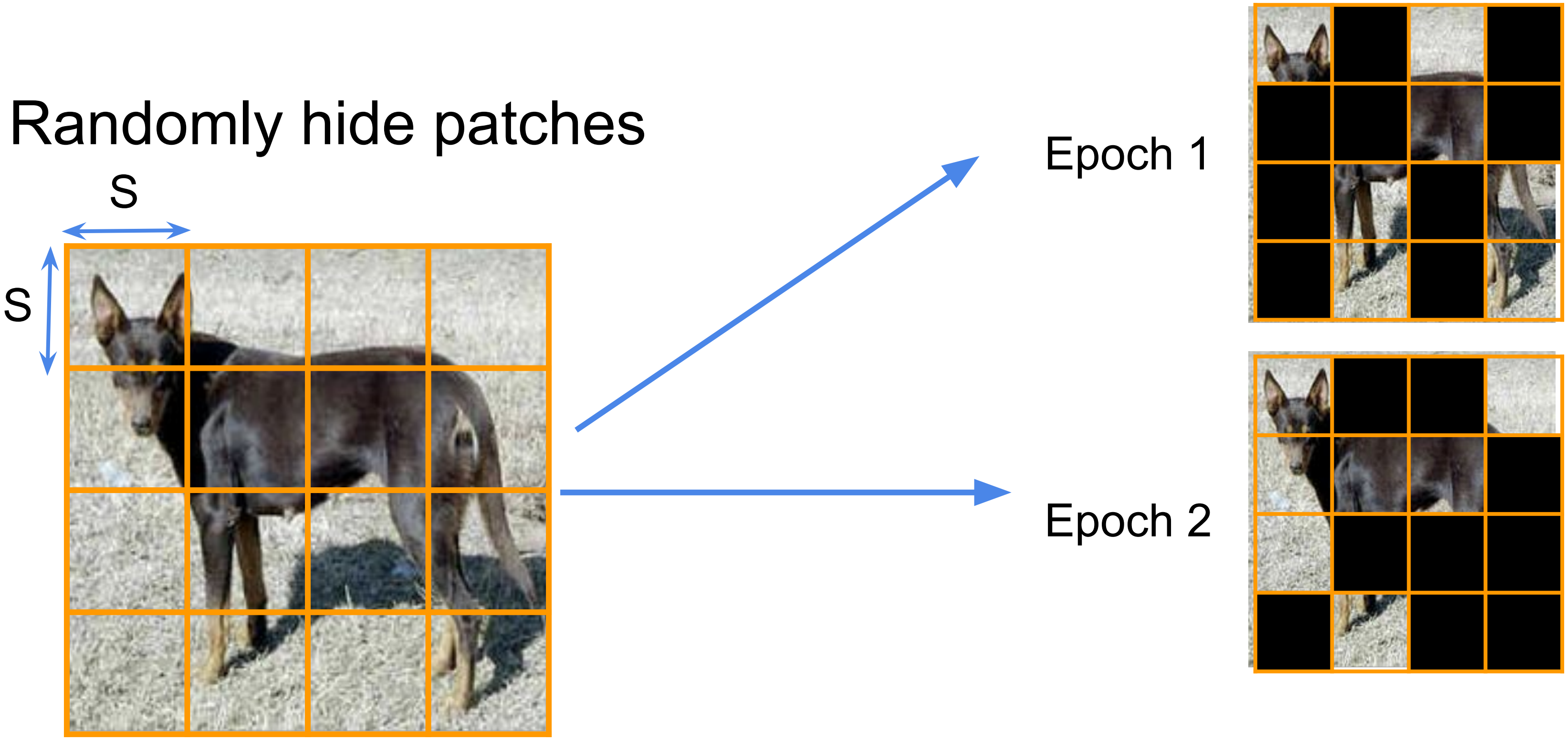
Training image
with label “dog”



Epoch 1

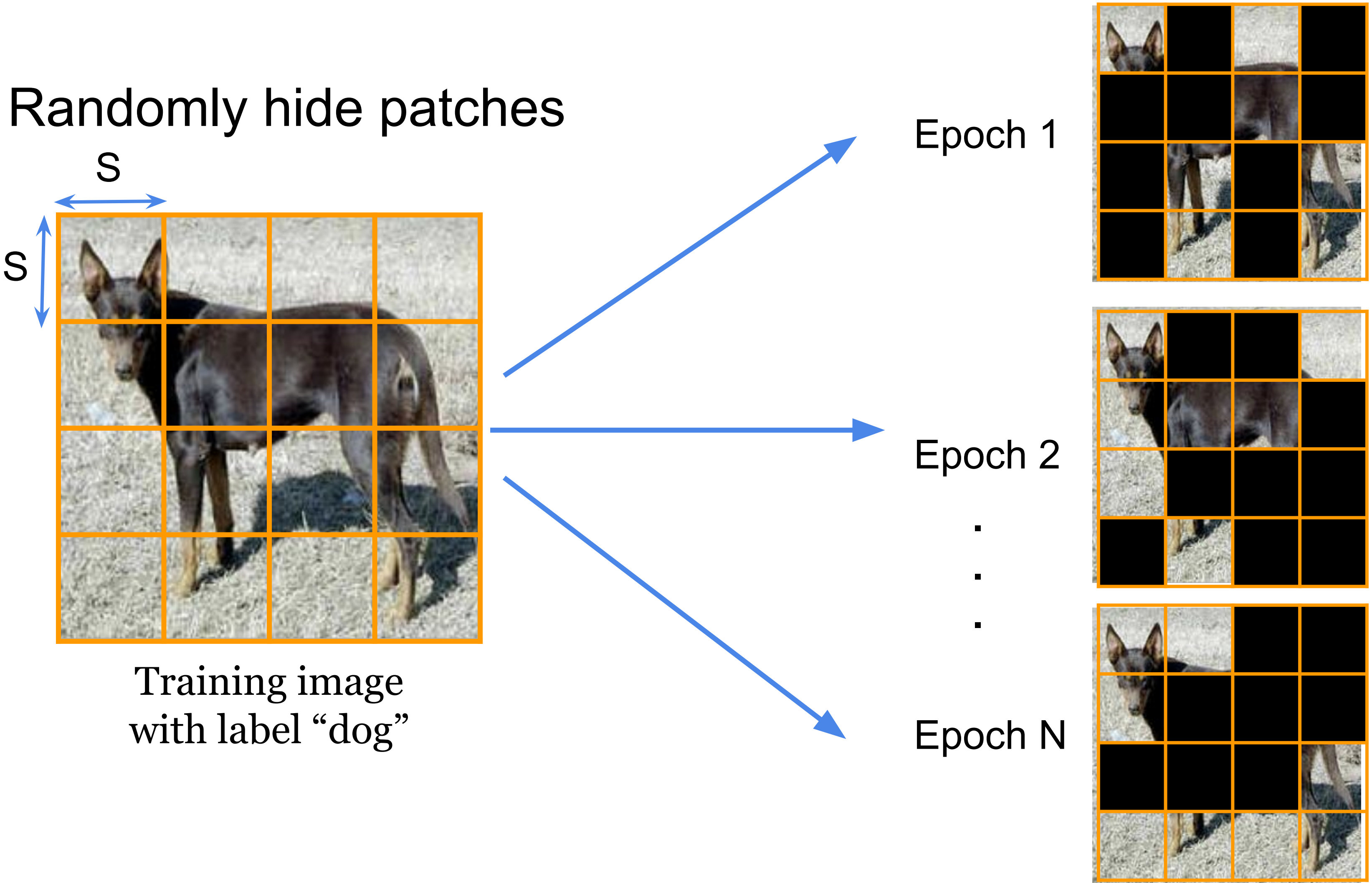


Randomly hide patches

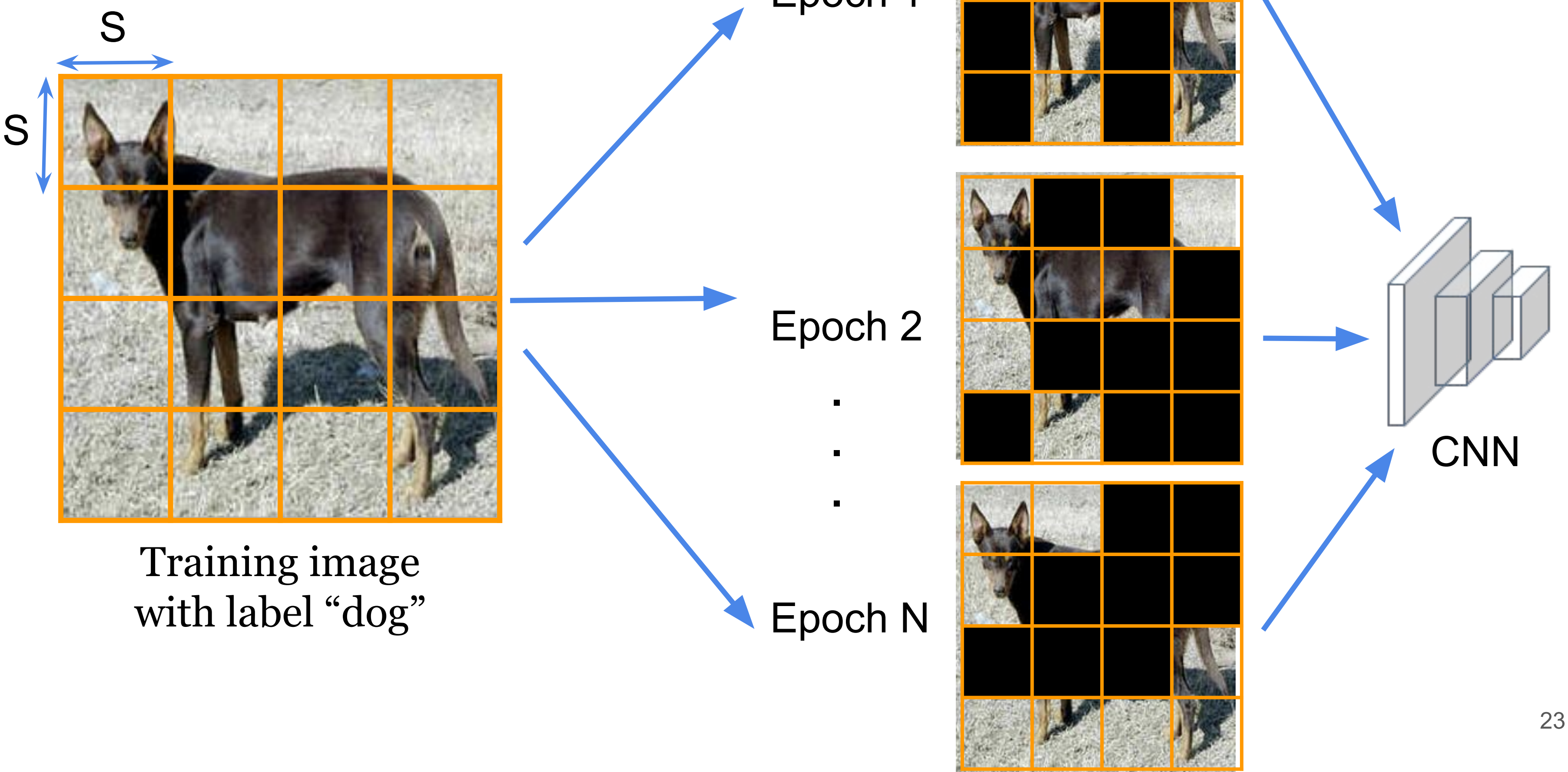


Training image
with label “dog”

Randomly hide patches



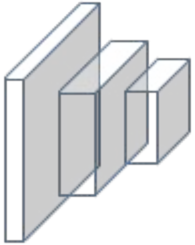
Feed each hidden image to image classification CNNs



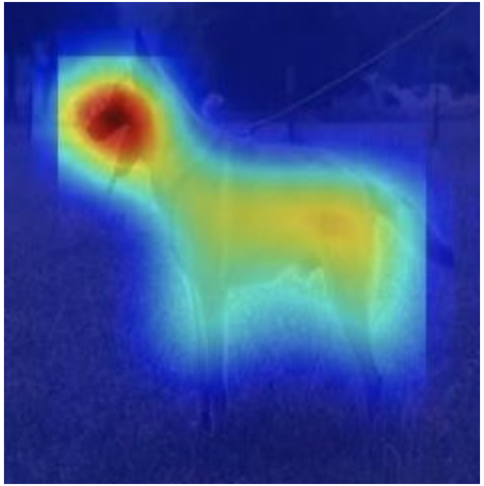
Training image with label "dog"

During testing feed full images into trained network

Test image

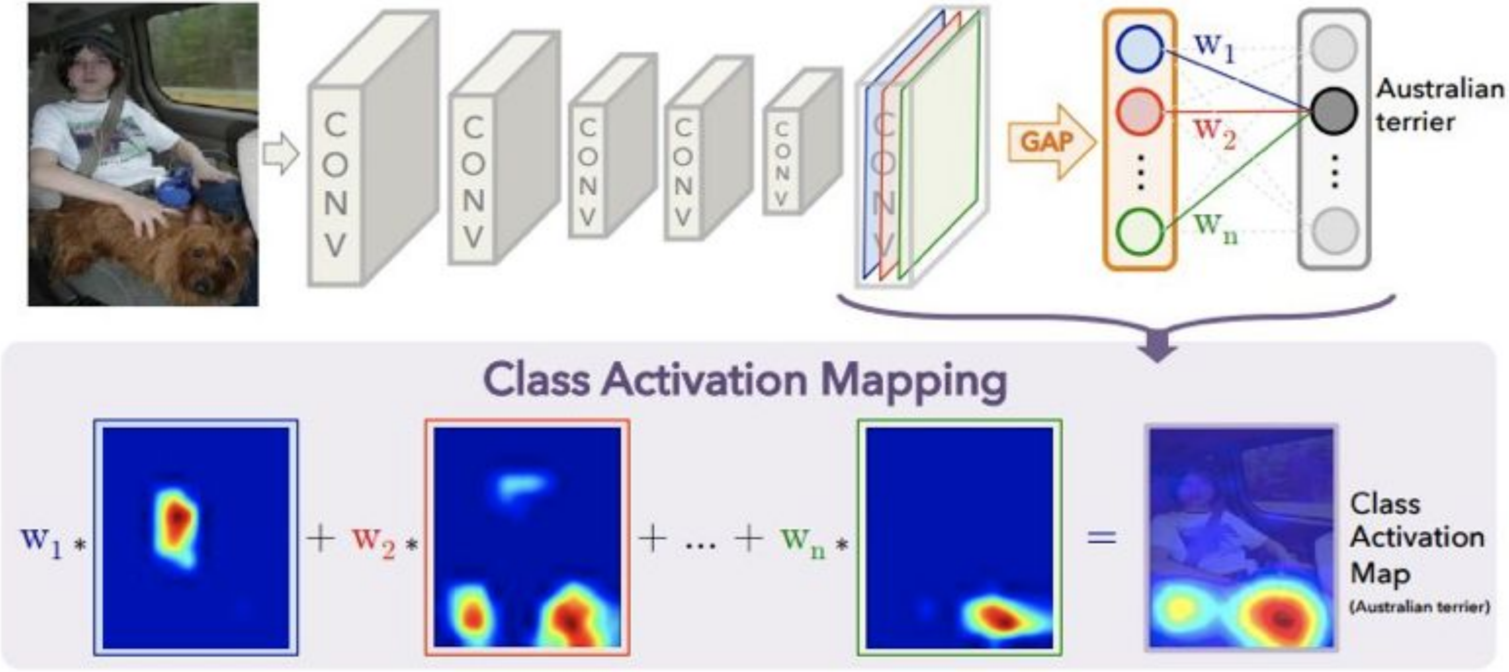


Trained CNN



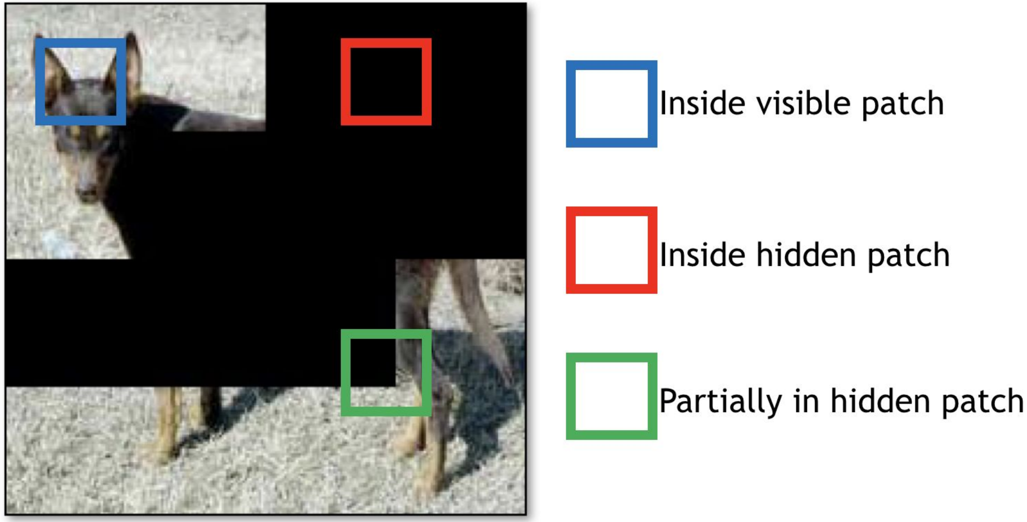
Class Activation Map
(CAM)
Predicted label: “dog”

Generating a Class Activation Map (CAM)



[Zhou et al. “Learning Deep Features for Discriminative Localization” CVPR 2016]

Setting the hidden pixel values



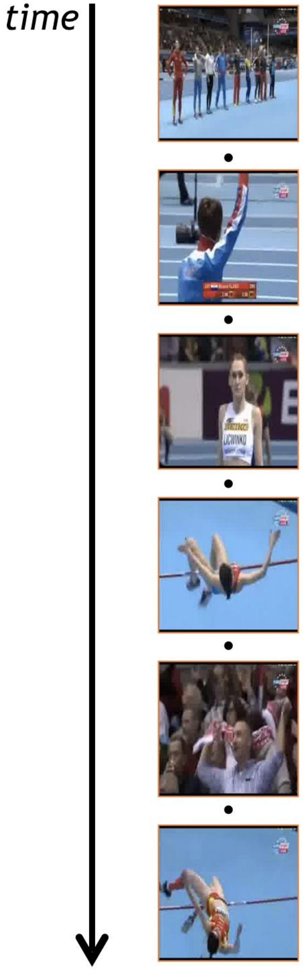
Set the RGB value of a hidden pixel to be the mean RGB vector of the **entire** dataset:

$$\mathbf{v} = \mu = \frac{1}{N_{pixels}} \sum_j \mathbf{x}_j$$

where **j** indexes all pixels in the entire training dataset and Npixel is the total number of pixels in the dataset.

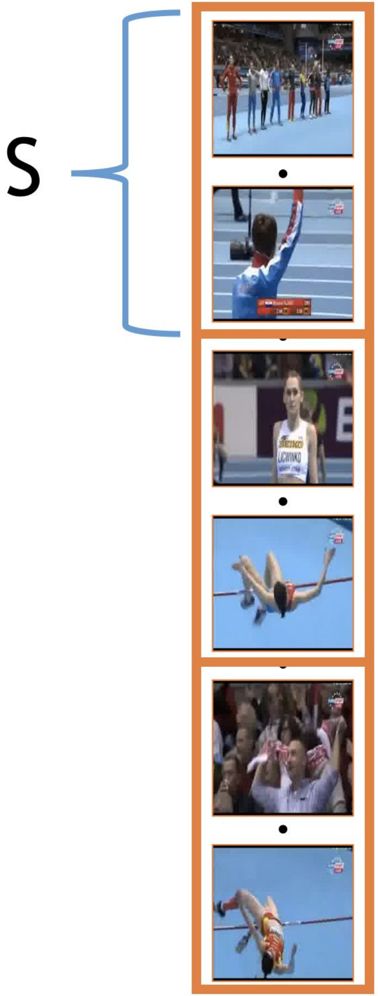
Hide-and-Seek (HaS) for:

- Weakly-supervised object localization in images
- Weakly-supervised temporal action localization in videos



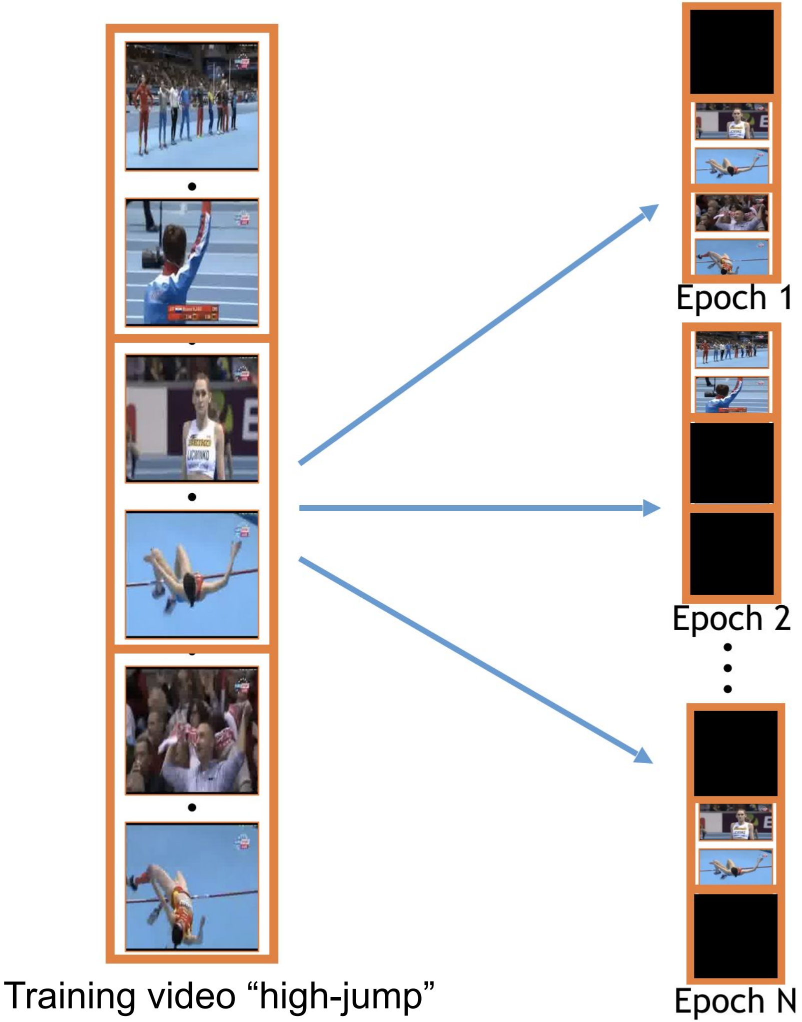
Training video “high-jump”

Divide training video into contiguous frame segments of size S



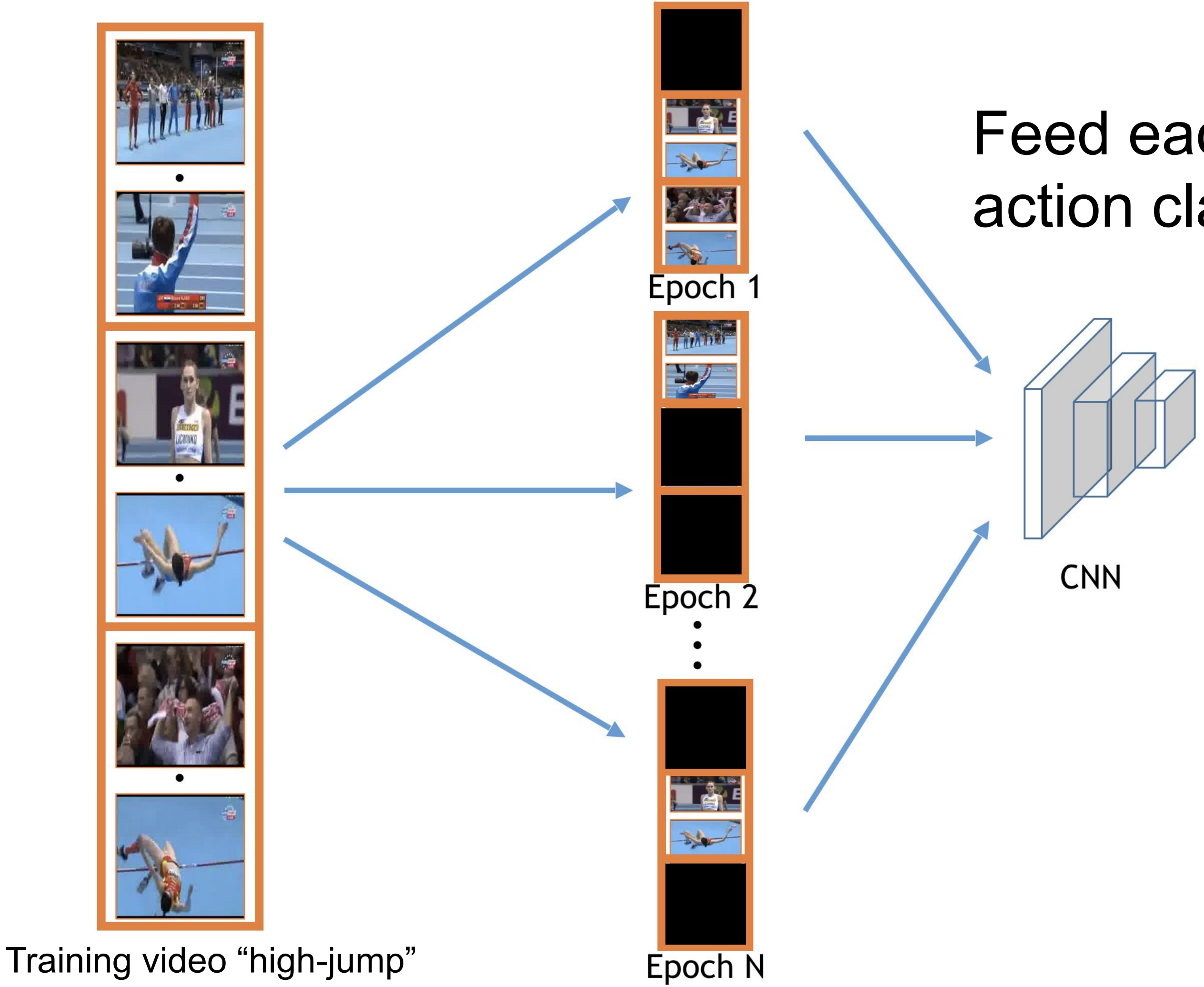
Training video “high-jump”

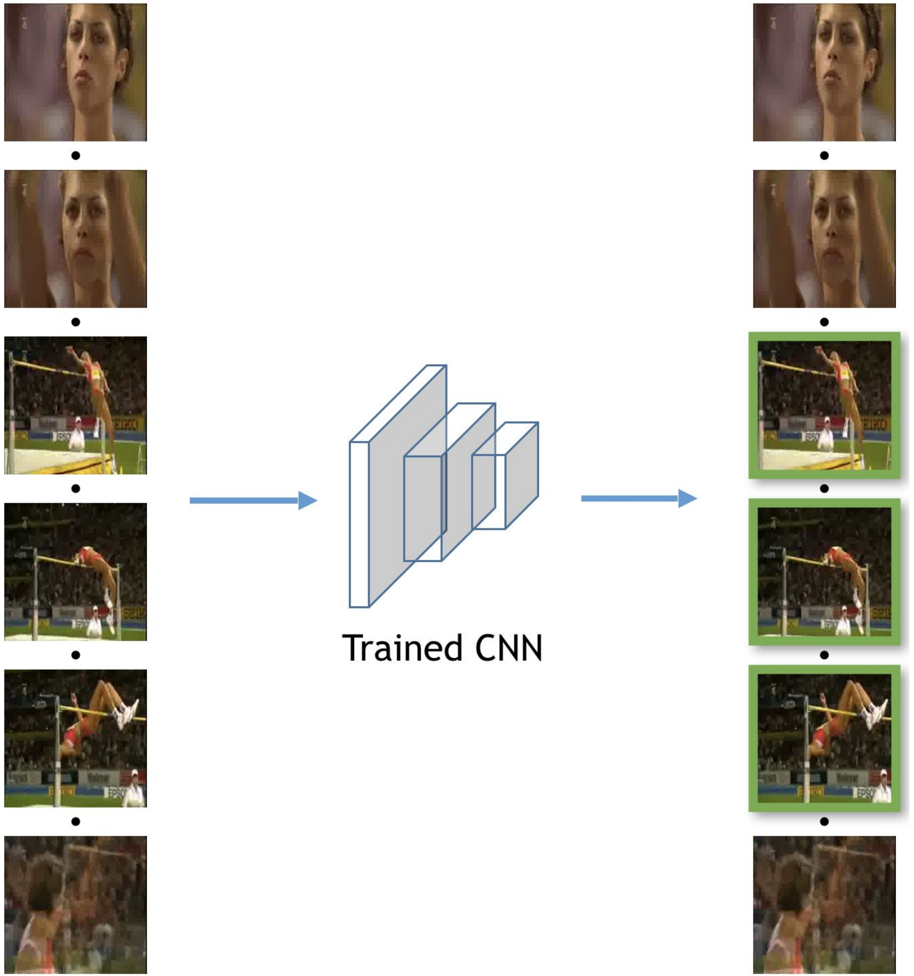
Divide training video into contiguous frame segments of size S



Feed each hidden video to action classification CNN

Feed each hidden video to action classification CNN





Training video “high-jump”

During testing feed full video into trained network

Other applications of HaS

- Weakly-supervised semantic segmentation
- Image classification
- Emotion recognition and age/gender estimation
- Person re-identification

[Singh, Krishna Kumar, et al. "Hide-and-Seek: A Data Augmentation Technique for Weakly-Supervised Localization and Beyond." arXiv preprint arXiv:1811.02545 (2018).]

Experiments and Results

Dataset

For object localization in images -

- ILSVRC 2016
- 1000 classes
- 1.2 million images with class labels for training

Dataset

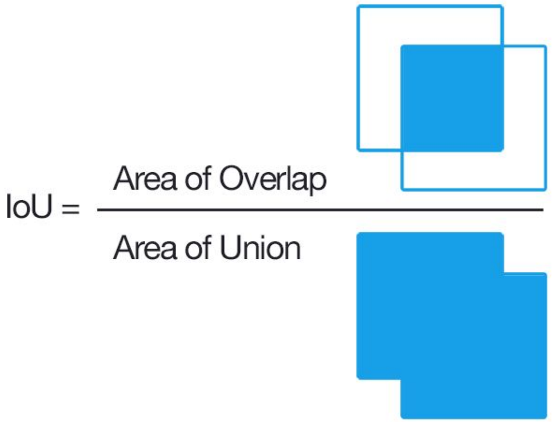
For action localization in video -

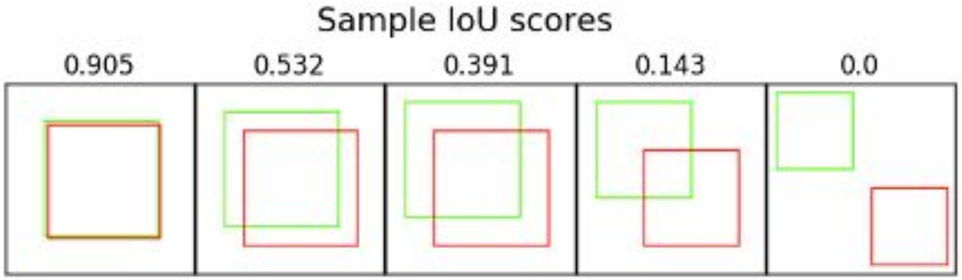
- THUMOS 2014 validation data
- 1010 untrimmed videos, 101 classes
- Train over all classes
- Evaluate 20 classes with temporal annotations
- Each video can contain multiple instances of a class

Metrics

For Object localization -

1) **Top-1 Loc:**
   Predicted class correct and bounding box > 50% IoU with ground truth
2) **GT-known Loc:**
   Bounding box > 50% IoU with ground truth of known class
3) **Top-1 Clas:**
   Classification accuracy



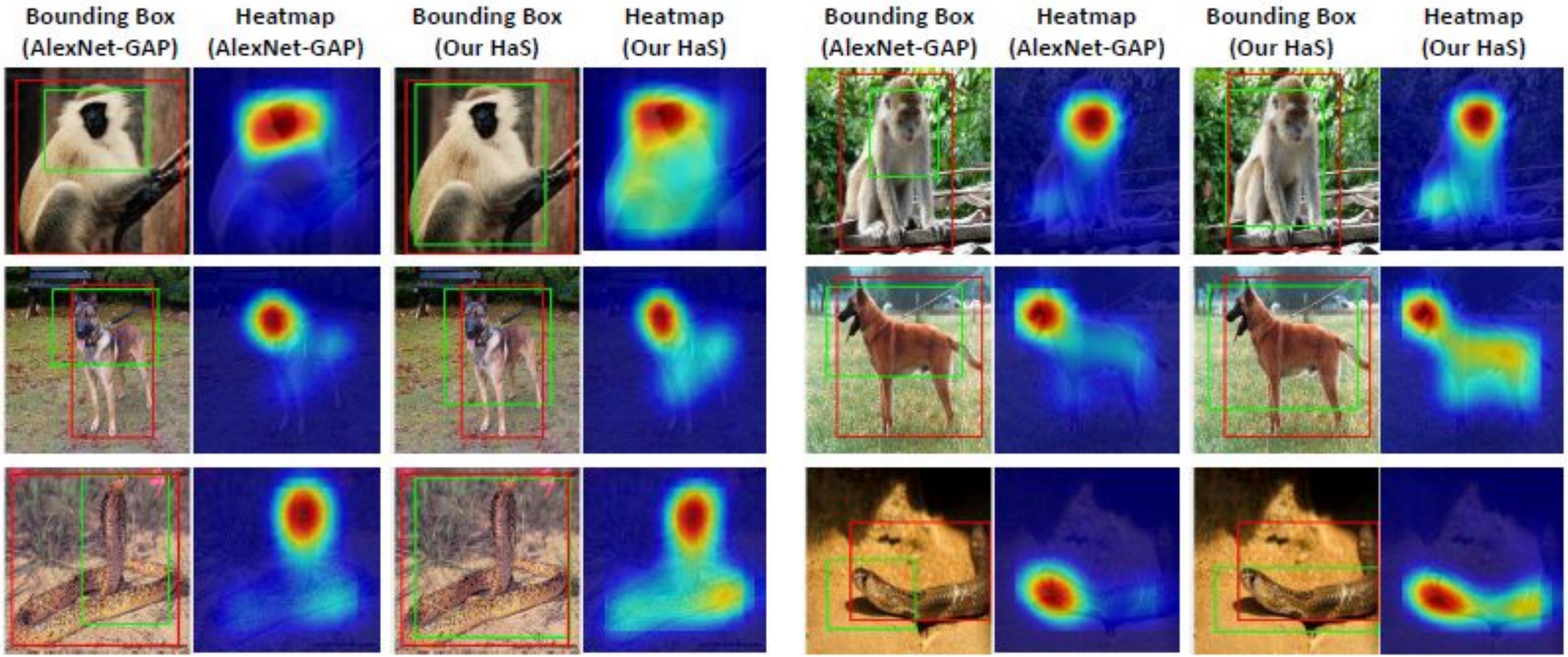


Metrics

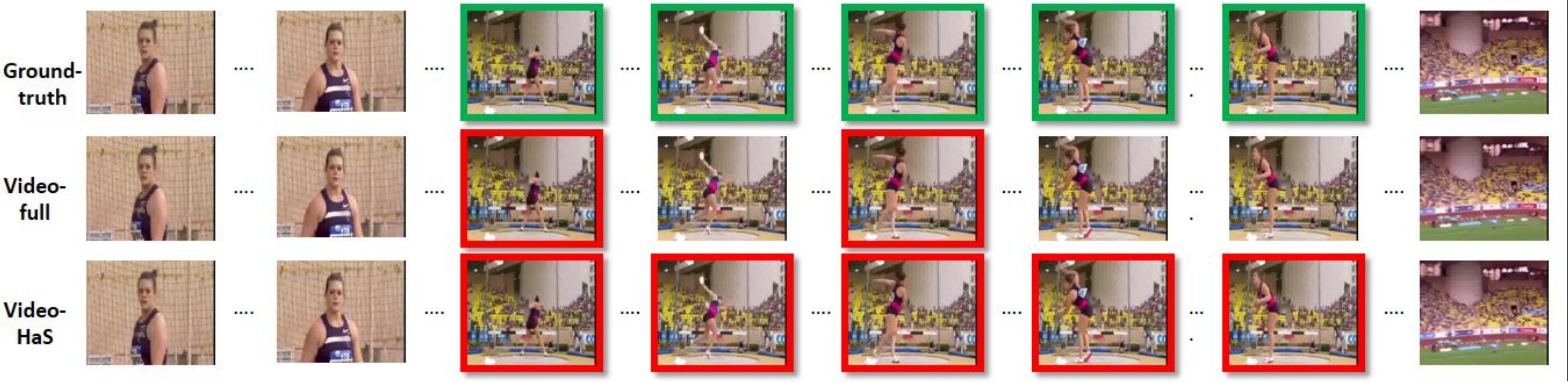
For action localization -

1) Mean average precision (mAP) for evaluation
2) Prediction is correct if IoU > $\boldsymbol{\theta}$
3) $\boldsymbol{\theta}$ = { 0.1, 0.2, 0.3, 0.4, 0.5 }
4) Assume ground truth class label is known

Qualitative results of object localization



Qualitative results of action localization



Slide courtesy of Krishna Kumar Singh, UC Davis

Qualitative results of action localization

Slide courtesy of Krishna Kumar Singh, UC Davis

Results of object localization

Methods	GT-known Loc	Top-1 Loc	Top-1 Clas
AlexNet-GAP [61]	54.90[2]	36.25	**60.23**
AlexNet-HaS-16	57.86	36.77	57.97
AlexNet-HaS-32	**58.75**	37.33	57.94
AlexNet-HaS-44	58.55	37.54	58.10
AlexNet-HaS-56	58.43	37.34	58.13
AlexNet-HaS-Mixed	58.68	**37.65**	58.68
GoogLeNet-GAP [61]	58.41[2]	43.60	**71.95**
GoogLeNet-HaS-16	59.83	44.62	70.49
GoogLeNet-HaS-32	**60.29**	**45.21**	70.70
GoogLeNet-HaS-44	60.11	44.75	70.34
GoogLeNet-HaS-56	59.93	44.78	70.37

Table 1. Localization accuracy on ILSVRC validation data with different patch sizes for hiding. Our Hide-and-Seek always performs better than AlexNet-GAP [61], which sees the full image.

Takeaway:

Randomly selecting the hidden patch size gives the best result.

Results of object localization

Methods	GT-known Loc	Top-1 Loc
Backprop on AlexNet [38]	-	34.83
AlexNet-GAP [61]	54.90	36.25
Ours	**58.68**	**37.65**
AlexNet-GAP-ensemble	56.91	38.58
Ours-ensemble	**60.14**	**40.40**
Backprop on GoogLeNet [38]	-	38.69
GoogLeNet-GAP [61]	58.41	43.60
Ours	**60.29**	**45.21**

Table 2. Localization accuracy on ILSVRC val data compared to state-of-the-art. Our method outperforms all previous methods.

Takeaway:

Averaging the CAM and class probabilities gives the best performance.

Results of object localization - Comparative Study

Methods	GT-known Loc	Top-1 Loc
Ours	**58.68**	**37.65**
AlexNet-dropout-trainonly	42.17	7.65
AlexNet-dropout-traintest	53.48	31.68

Table 3. Our approach outperforms Dropout [44] for localization.

Takeaway: HaS method much better at improving localization performance than dropout.

Results of object localization - Comparative Study

Methods	GT-known Loc	Top-1 Loc
AlexNet-GAP	54.90	36.25
AlexNet-Avg-HaS	58.43	37.34
AlexNet-GMP	50.40	32.52
AlexNet-Max-HaS	**59.27**	**37.57**

Table 4. Global average pooling (GAP) vs. global max pooling (GMP). Unlike [61], for Hide-and-Seek GMP still performs well for localization. For this experiment, we use patch size 56.

Takeaway: GMP performs better as HaS already trains network for better localization.

Results of object localization - Comparative Study

Methods	GT-known Loc	Top-1 Loc
AlexNet-GAP	54.90	36.25
AlexNet-HaS-conv1-5	57.36	36.91
AlexNet-HaS-conv1-11	**58.33**	**37.38**

Table 5. Applying Hide-and-Seek to the first conv layer. The improvement over [61] shows the generality of the idea.

Takeaway: HaS method intuition can be applied to even convolution layer filter outputs to give the same boost in performance.

Results of object localization - Comparative Study

Methods	GT-known Loc	Top-1 Loc
AlexNet-HaS-25%	57.49	37.77
AlexNet-HaS-33%	58.12	38.05
AlexNet-HaS-50%	58.43	37.34
AlexNet-HaS-66%	58.52	35.72
AlexNet-HaS-75%	58.28	34.21

Table 6. Varying the hiding probability. Higher probabilities lead to decrease in *Top-1 Loc* whereas lower probability leads to smaller *GT-known Loc*. For this experiment, we use patch size 56.

Takeaway: There is a trade off between localization and callsification accuracy wrt hiding probability.

Results of action localization

Methods	IOU thresh = 0.1	0.2	0.3	0.4	0.5
Video-full	34.23	25.68	17.72	11.00	6.11
Video-HaS	**36.44**	**27.84**	**19.49**	**12.66**	**6.84**

Table 7. Action localization accuracy on THUMOS validation data. Across all 5 IoU thresholds, our Video-HaS outperforms the full video baseline (Video-full).

Classification results for higher capacity network

	CIFAR-10			CIFAR-100			ImageNet
	ResNet44	ResNet56	ResNet110	ResNet44	ResNet56	ResNet110	ResNet50
Full	94.19	94.66	94.87	74.37	75.24	77.44	76.15
HaS	**94.97**	**95.41**	**95.53**	**75.82**	**76.47**	**78.13**	**77.20**

"Hide-and-Seek: A Data Augmentation Technique for Weakly-Supervised Localization and Beyond" (Singh, 2018).

Using Hide-and-Seek as data augmentation improves performance of various vision tasks

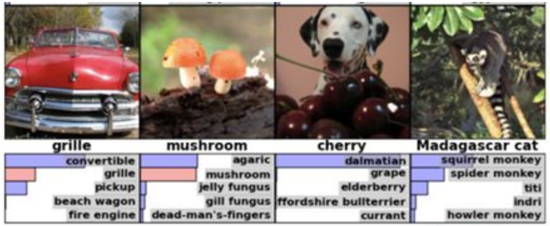


Image classification +1.1%
[He et al. 2015]

Semantic segmentation +1.3%
[Long et al. 2015]

Face recognition tasks +1%
(emotion, age, gender) [Khorrami et al. 2015]

Person Reidentification +1.6%
[Zhong et al. 2018]

Slide courtesy of Krishna Kumar Singh, UC Davis

Our approach improves image classification when objects are partially-visible

Ground-truth: African Crocodile
AlexNet-GAP: Trilobite
Ours: African Crocodile

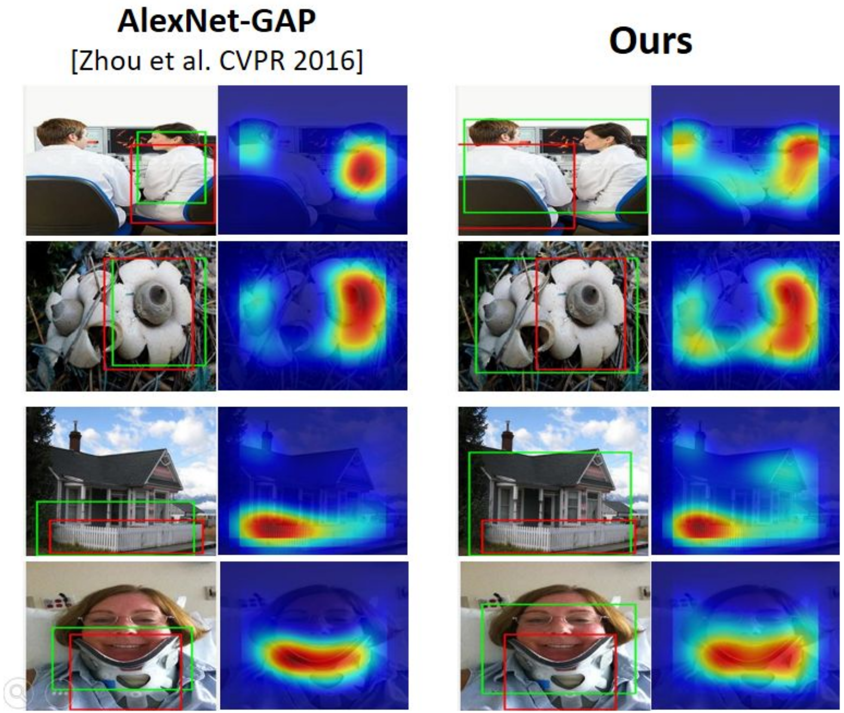
Ground-truth: Electric Guitar
AlexNet-GAP: Banjo
Ours: Electric Guitar

Ground-truth: Notebook
AlexNet-GAP: Waffle Iron
Ours: Notebook

Ground-truth: Ostrich
AlexNet-GAP: Border Collie
Ours: Ostrich

Slide courtesy of Krishna Kumar Singh, UC Davis

Fail Cases



Merging spatially-close instances together

Localizing co-occurring context

Slide courtesy of Krishna Kumar Singh, UC Davis

Fail Cases

Our approach can fail by localizing co-occurring context

Slide courtesy of Krishna Kumar Singh, UC Davis

Strength and weakness

Strength -

1) The Hide and Seek method can be applied to any architecture
2) Better than Dropout for localization problem
3) Can be used as form of data augmentation to improve other tasks like segmentation, face recognition, etc.

Weakness -

1) The classification accuracy decreases for lower capacity networks
2) Spatially close instances and co-occurring contexts cause method to fail
3) Current method will not suffice for videos with multiple action labels

Future work

1) The patch size and hiding probabilities are hyper-parameters.
2) Dynamically learn patch size and hiding probability during training.

Thank You

Any Questions?

Implementation - Object localization

Models	Conv layer	Train epochs	Learning rate	Batch Norm	Cam Threshold
AlexNet-GAP	512, (3x3), stride = 1, pad =1	55	0.01 → 0.0001	Yes	20%
GoogLeNet-GAP	1024, (3x3), stride = 1, pad =1	40	0.01 → 0.0001	Yes	30%

Implementation - Action localization

Sample 2000 features @ 10 feats/sec

Model trained on Sports 1 Million	Compute C3D fc7 Features

Segment video

$F_{segment} = 100$, $P_{hide} = 0.5$	20 equal length segments

Feed C3D features to CNN

500 kernels, (1x1), stride = 1	Mean C3D for hidden frames

Prediction

Threshold = 50% of max CAM	All continuous segments after thresholding