Weakly-Supervised Semantic Segmentation using End-to-End Adversarial Erasing

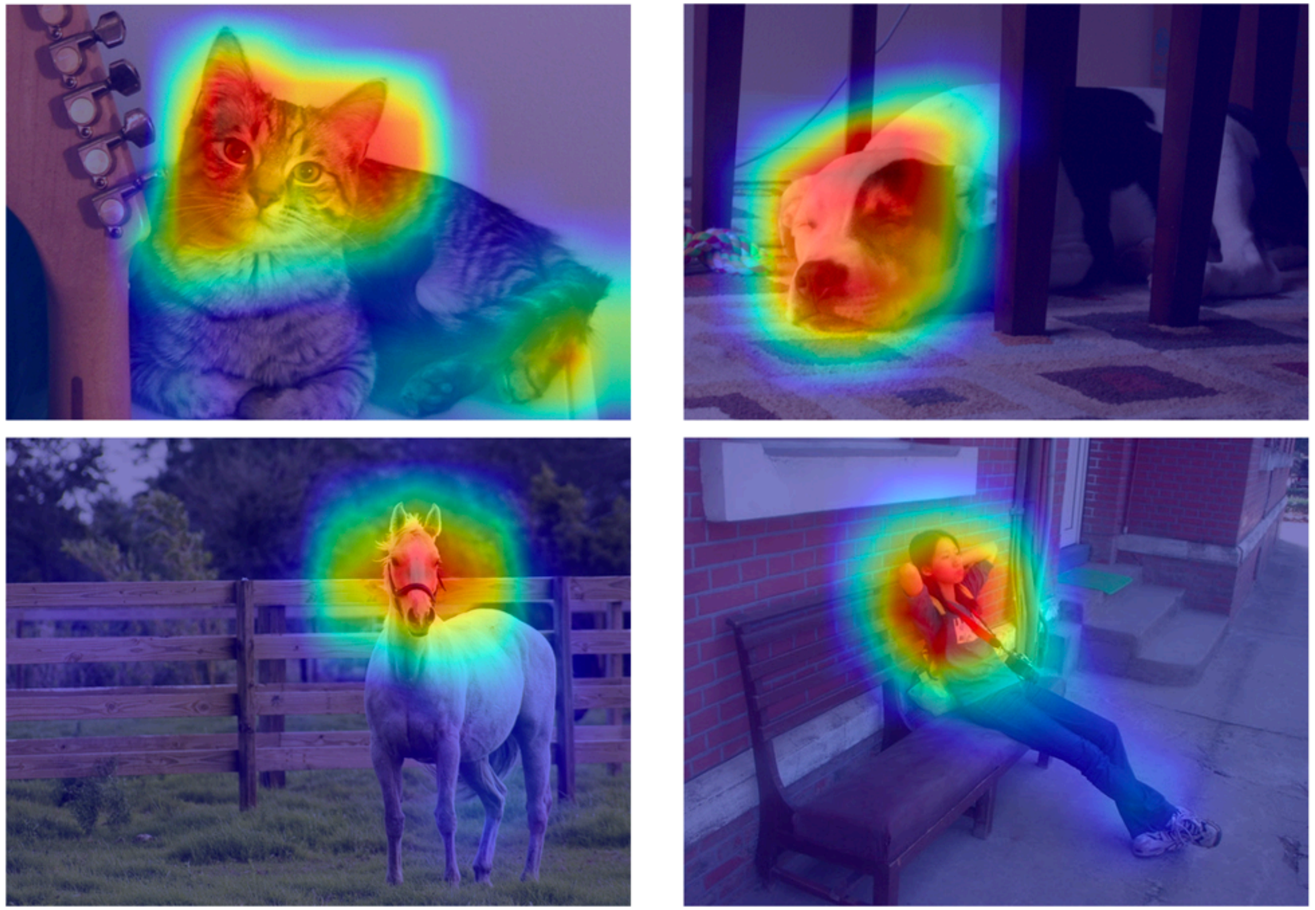
Erik Stammes 21/05/2021

Semantic Segmentation

Types of supervision

Attention Maps

Provide object localization



Adversarial Erasing

Iterative erasing of the attention map

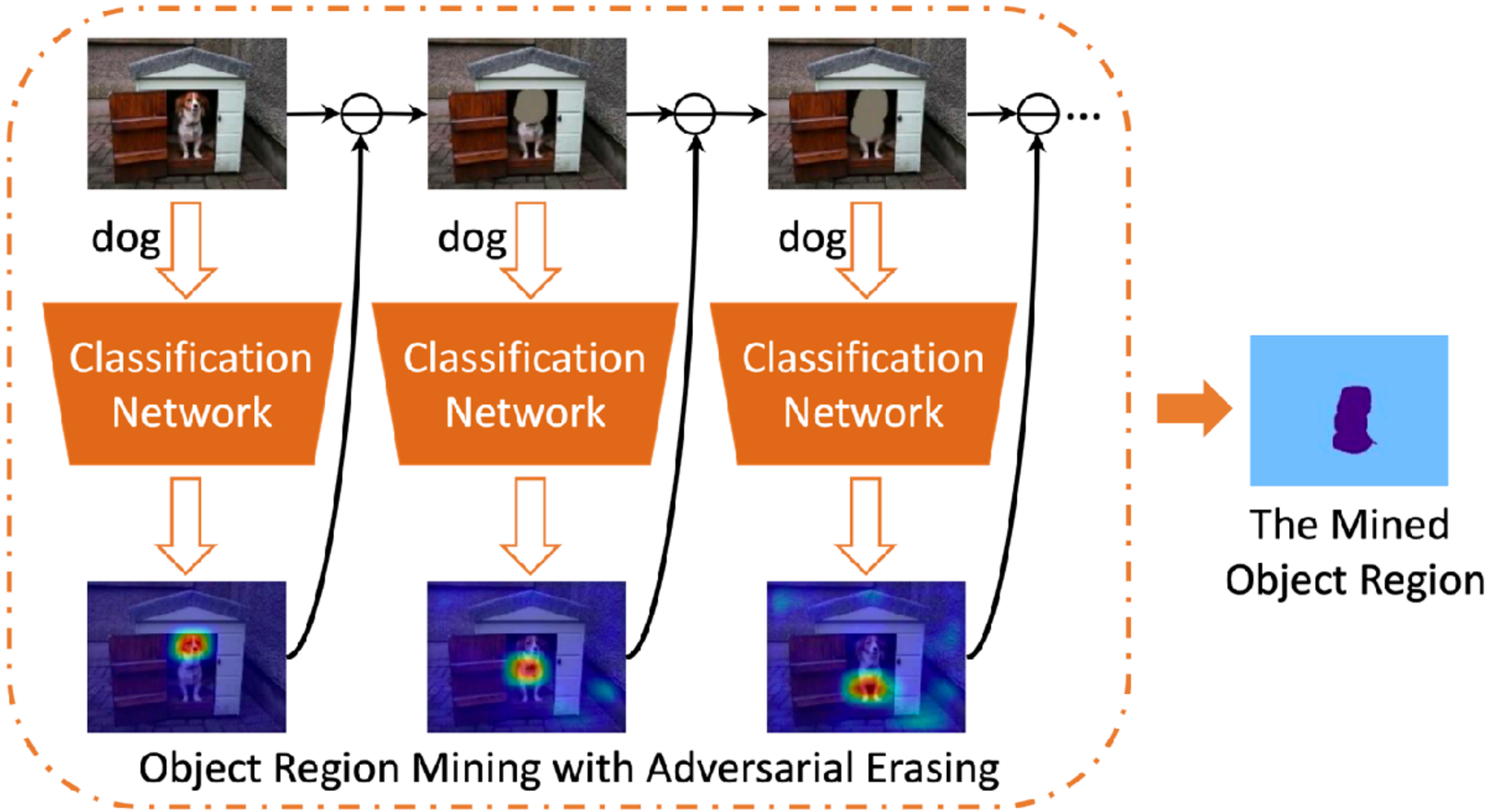


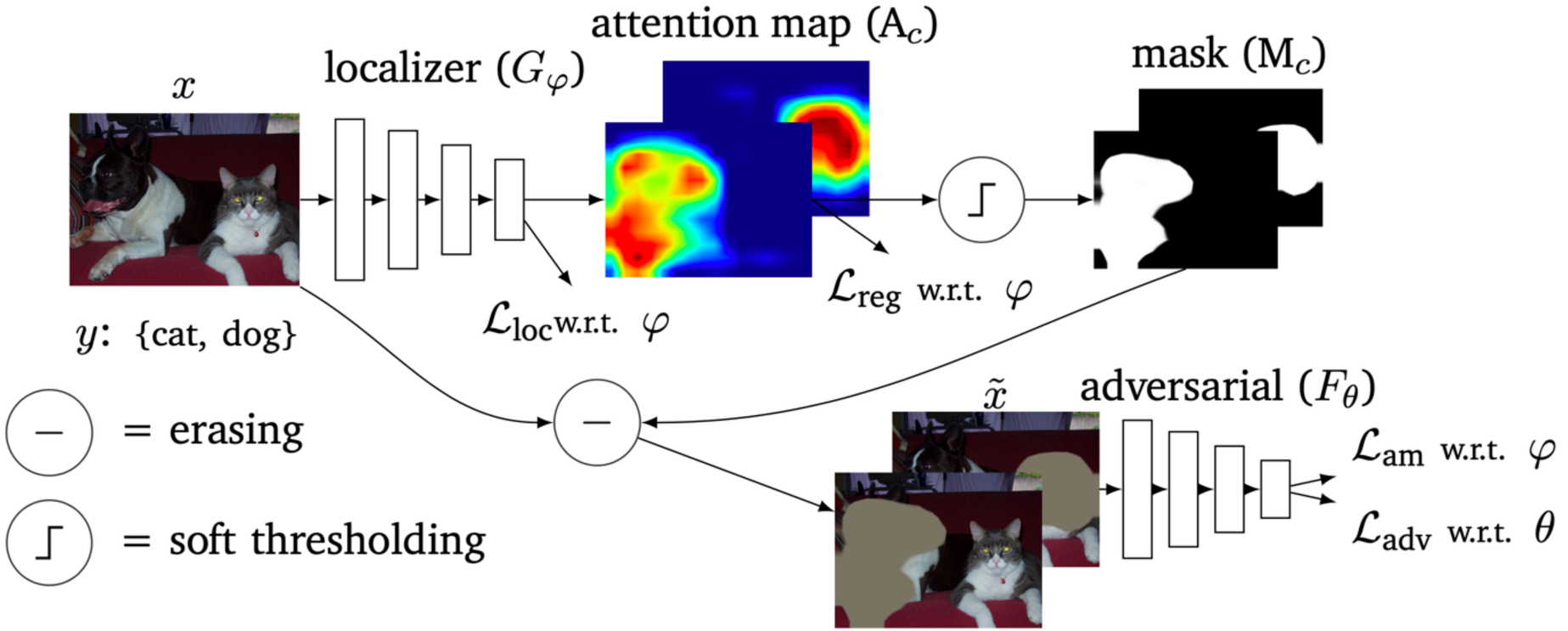
Image from: Wei, Yunchao, et al. "Object region mining with adversarial erasing: A simple classification to semantic segmentation approach." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

Issues with Adversarial Erasing

- Multiple training and inference steps
- Fusion of attention maps
- Weight sharing
- Saliency estimation methods
- Bloated with bells and whistles
- “Adversarial”

End-to-End Adversarial Erasing

Our proposed approach



$$\mathcal{L}_{\text{loc}}\left(G_\varphi(x_i), y_i\right) = -\frac{1}{C}\sum_c y_{i,c} \ln\left(G_\varphi(x_i)\right) + \left(1 - y_{i,c}\right) \ln\left(1 - G_\varphi(x_i)\right)$$

$$\mathcal{L}_{\text{adv}}\left(F_\theta(\tilde{x}_{i,c}), y_i\right) = -\frac{1}{C}\sum_c y_{i,c} \ln\left(F_\theta(\tilde{x}_{i,c})\right) + \left(1 - y_{i,c}\right) \ln\left(1 - F_\theta(\tilde{x}_{i,c})\right)$$

$$\mathcal{L}_{\text{am}}(\tilde{x}_i, y_i) = \frac{1}{C}\sum_{c \in y_i} F_\theta(\tilde{x}_{i,c})$$

$$\mathcal{L}_{\text{reg}}(x_i, y_i) = \frac{1}{W \times H \times C}\sum_{c \in y_i}\sum_{j,k} A_c(x_i)_{j,k},$$

Issues with Adversarial Erasing

And how we resolve them

- Multiple training and inference steps
- Fusion of attention maps
- Saliency estimation methods
- Weight sharing
- Bloated with bells and whistles
- “Adversarial”

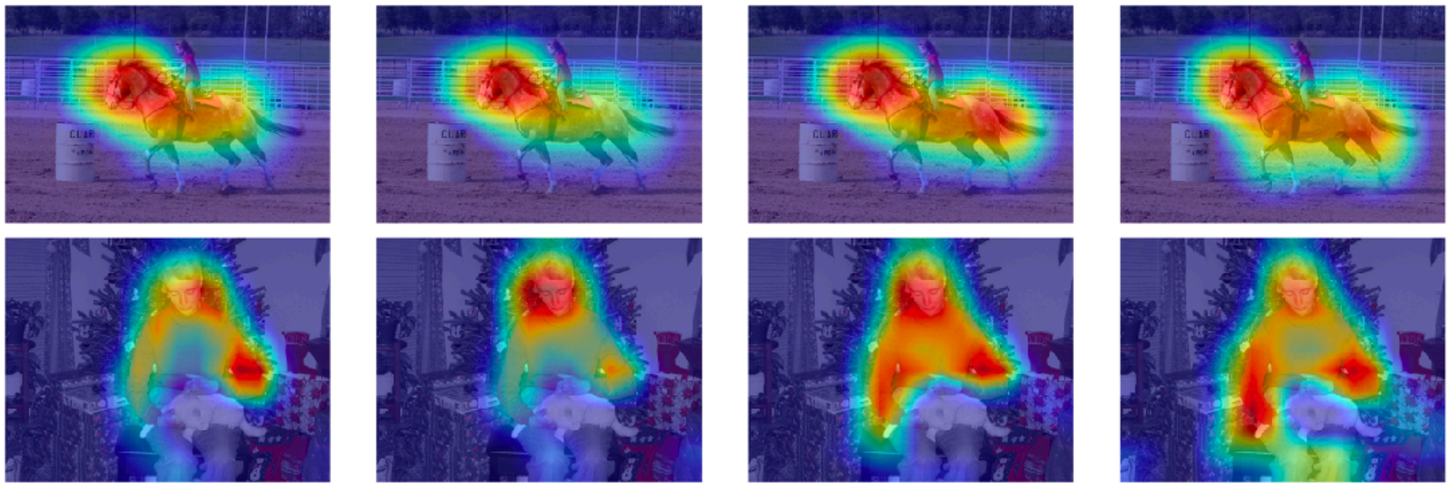
- Trained end-to-end
- Learnable attention map
- Regularization loss
- Two distinctly trained networks
- Simple and integrable
- Truly adversarial

Results

Segmentation performance

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{loc}} + \alpha\mathcal{L}_{\text{am}} + \beta\mathcal{L}_{\text{reg}}$$

α	mIoU	Precision	Recall
0	41.37 $\pm$ 0.26	58.26 $\pm$ 0.50	58.18 $\pm$ 0.66
0.01	42.51 $\pm$ 0.41	57.79 $\pm$ 0.72	60.88 $\pm$ 0.52
0.05	**43.89** $\pm$ 0.40	54.78 $\pm$ 1.03	68.13 $\pm$ 1.51
0.1	42.88 $\pm$ 0.99	52.68 $\pm$ 2.30	69.31 $\pm$ 1.84



$\alpha = 0$ $\alpha = 0.01$ $\alpha = 0.05$ $\alpha = 0.1$

Integrability

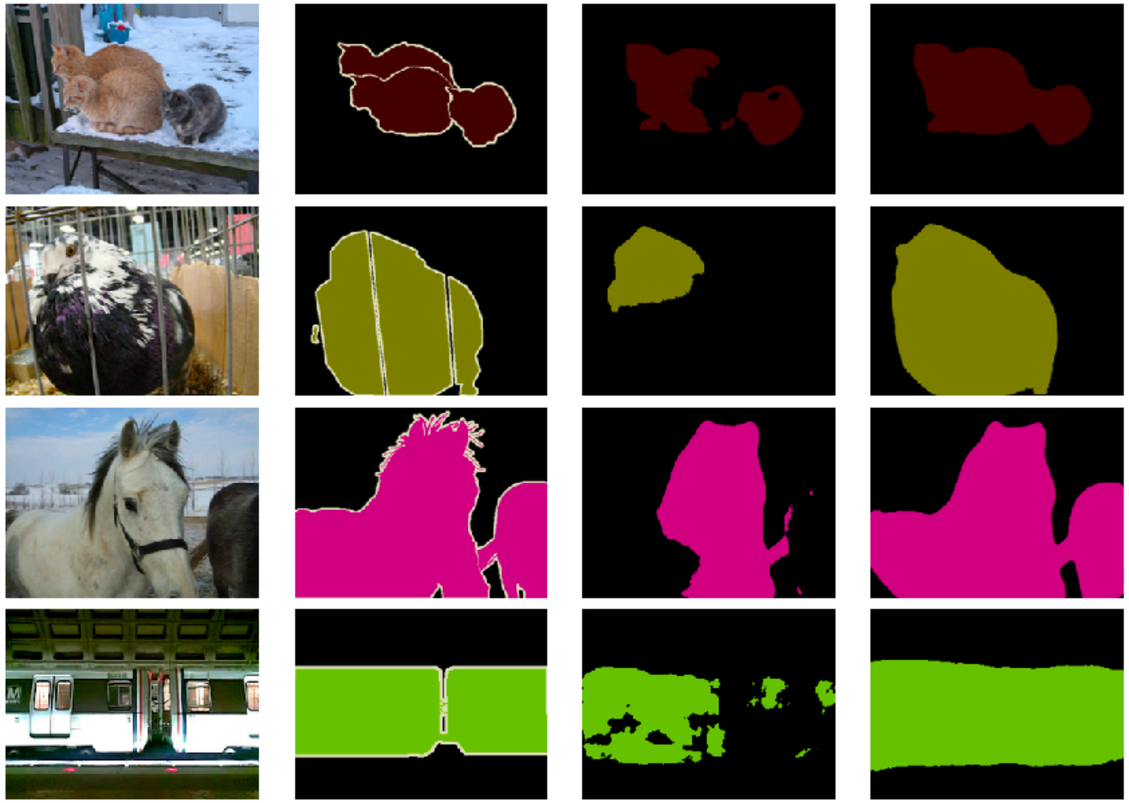
Pixel-level Semantic Affinity (PSA)
End-to-End Adversarial Erasing (EADER)

	CAM			AffinityNet	DeepLabV3+
Model	mIoU	Precision	Recall	mIoU	mIoU
PSA	46.8	60.3	66.7	58.7	60.7
PSA w/ EADER	**48.6**	**61.3**	**68.7**	**60.1**	**62.8**

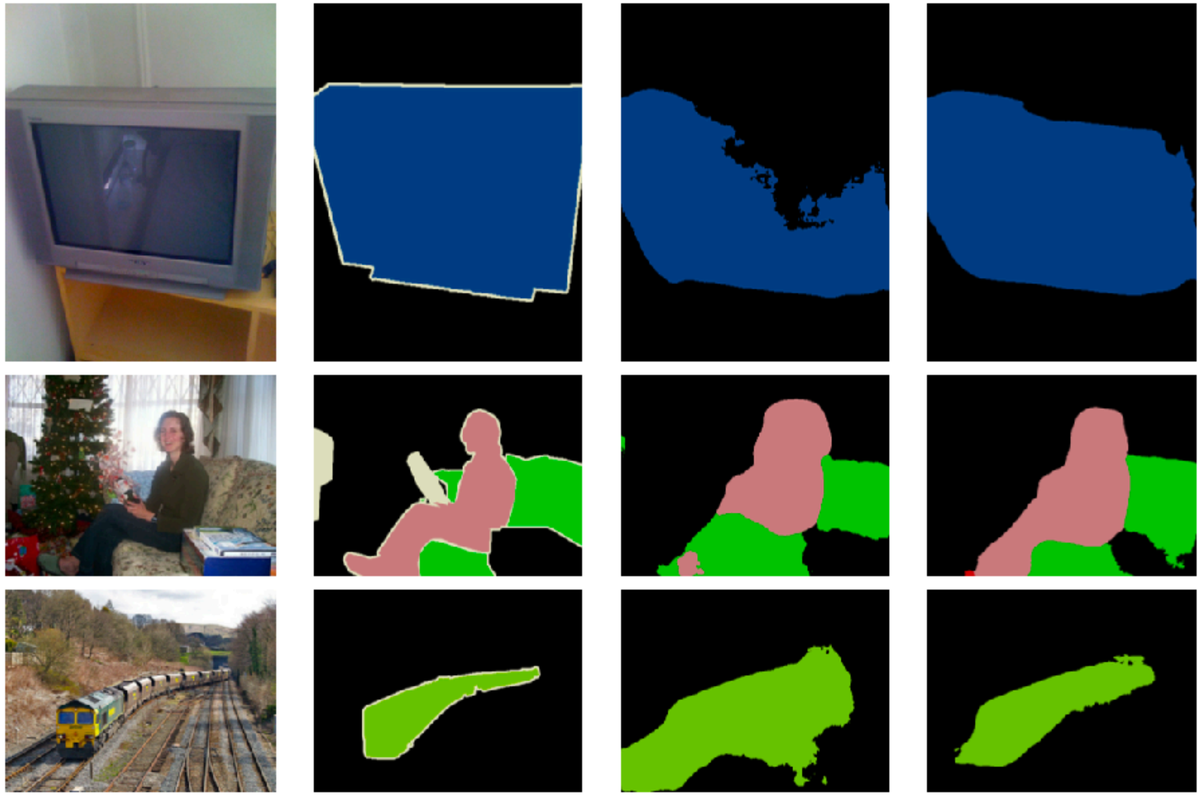
	Validation	
Class	PSA	EADER
background	86.7	**88.2**
aeroplane	53.2	**54.9**
bike	29.1	**31.3**
bird	76.7	**84.1**
boat	44.2	**58.2**
bottle	67.7	**70.9**
bus	**85.2**	83.0
car	72.4	**76.2**
cat	71.7	**82.1**
chair	**26.7**	24.4
cow	76.5	**80.6**
dining table	**40.9**	35.8
dog	72.2	**80.7**
horse	68.2	**76.4**
motorbike	70.2	**73.7**
person	66.4	**70.8**
potted plant	**37.8**	15.4
sheep	**80.9**	77.2
sofa	**38.5**	34.6
train	62.8	**66.4**
tv	45.4	**52.6**
mean	60.7	**62.8**

Visual Results

Segmentation masks



Input Ground Truth PSA PSA w/ EADER



Input Ground Truth PSA PSA w/ EADER

Comparison

To Adversarial Erasing methods

Method	Supervision	Validation	Test
AE-PSL[17]	$\mathcal{I}+\mathcal{S}$	55.0	55.7
GAIN [20]	$\mathcal{I}+\mathcal{S}$	55.3	56.8
DCSP [85]	$\mathcal{I}+\mathcal{S}$	60.8	61.9
SeeNet [22]	$\mathcal{I}+\mathcal{S}$	**63.1**	62.8
ACoL [21]	$\mathcal{I}$	56.1	-
EADER (Ours)	$\mathcal{I}$	62.8	**63.8**

[17] IWei, Yunchao, et al. "Object region mining with adversarial erasing: A simple classification to semantic segmentation approach." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

[20] Li, Kunpeng, et al. "Tell me where to look: Guided attention inference network." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

[21] Zhang, Xiaolin, et al. "Adversarial complementary learning for weakly supervised object localization." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

[22] Hou, Qibin, et al. "Self-erasing network for integral object attention." *Advances in Neural Information Processing Systems*. 2018.

[85] Chaudhry, Arslan, Puneet K. Dokania, and Philip HS Torr. "Discovering Class-Specific Pixels for Weakly-Supervised Semantic Segmentation."

Conclusion

End-to-End Adversarial Erasing

- A simple formulation that is agnostic to neural network architecture, attention map generation method and does not require saliency masks
- Easily integrable into existing methodologies
- Outperforms all existing adversarial erasing methods
- Outperforms many existing weakly-supervised semantic segmentation methods and even some fully-supervised ones.
- We hypothesize that better performance can be obtained by integrating it into a better performing baseline

Thanks!

Code available online: https://github.com/ErikStammes/EADER

Algorithm 1: End-to-end adversarial erasing (EADER)

Data: Training set $\mathcal{D} = \{x_i, y_i\}_{i=1}^{N}$, parameters α, β
Result: Segmentation masks S
while *training has not converged* **do**
  Forward image through localizer $\hat{y}_i^{\text{loc}} \leftarrow G_\varphi(x_i)$;
  Generate attention map $a_{i,c} \leftarrow A_c(x_i)$;
  Generate mask $m_{i,c} \leftarrow M_c(x_i)$;
  Erase mask from image $\hat{x}_{i,c} \leftarrow \text{erase}(x_i, m_{i,c})$;
  Forward erased image through adversarial $\hat{y}_i^{\text{adv}} \leftarrow F_\theta(\hat{x}_{i,c})$;
  if *train localizer* **then**
    Compute localizer loss $\mathcal{L}_{\text{loc}} \leftarrow \mathcal{L}_{\text{loc}}(\hat{y}_i^{\text{loc}}, y_i)$;
    Compute attention mining loss $\mathcal{L}_{\text{am}} \leftarrow \mathcal{L}_{\text{am}}(\hat{y}_i^{\text{adv}}, y_i)$;
    Compute regularization loss $\mathcal{L}_{\text{reg}} \leftarrow \mathcal{L}_{\text{reg}}(a_{i,c})$;
    Compute total loss $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{loc}} + \alpha\mathcal{L}_{\text{am}} + \beta\mathcal{L}_{\text{reg}}$;
    Update φ w.r.t $\mathcal{L}_{\text{total}}$;
  else
    Compute adversarial loss $\mathcal{L}_{\text{adv}} \leftarrow \mathcal{L}_{\text{adv}}(\hat{y}_i^{\text{adv}}, y_i)$;
    Update θ w.r.t $\mathcal{L}_{\text{adv}}$;
  end
end
for $i \leftarrow 1$ **to** N **do**
  Generate segmentation mask $S_i \leftarrow \arg\max(a_i)$;
end

Method	Feature Extractor	Fully Supervised Model (Backbone)	Supervision	Validation	Test
FCN [28]	-	(VGG16)	$\mathcal{F}$	-	62.2
WideResNet-38 [30]	-	(WideResNet-38)	$\mathcal{F}$	80.8	82.5
PSPNet [5]	-	(ResNet-101)	$\mathcal{F}$	-	82.6
DeepLabV3+ [4]	-	(Xception-65)	$\mathcal{F}$	84.6	87.8
SN_B [53]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	41.9	43.2
AE-PSL[17]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	55.0	55.7
Oh *et al.* [55]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	55.7	56.7
GAIN [20]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	55.3	56.8
MCOF [64]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	56.2	57.6
DCSP [85]	VGG-16	-	$\mathcal{I}+\mathcal{S}$	58.6	59.2
DSRG [47]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	59.0	60.4
SeeNet [22]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	61.1	60.7
MDC [52]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	60.4	60.8
MCOF [64]	VGG-16	DeepLab (ResNet-101)	$\mathcal{I}+\mathcal{S}$	60.3	61.2
DCSP [85]	ResNet-101	-	$\mathcal{I}+\mathcal{S}$	60.8	61.9
FickleNet [49]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	61.2	61.9
Fan *et al.* [50]	ResNet-50	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	61.3	62.1
SeeNet [22]	VGG-16	DeepLab (ResNet-101)	$\mathcal{I}+\mathcal{S}$	63.1	62.8
OAA+ [51]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}+\mathcal{S}$	63.1	62.8
DSRG [47]	VGG-16	DeepLab (ResNet-101)	$\mathcal{I}+\mathcal{S}$	61.4	63.2
Fan *et al.* [50]	ResNet-50	DeepLab (ResNet-101)	$\mathcal{I}+\mathcal{S}$	63.6	64.5
CIAN [54]	VGG-16	DeepLab (ResNet-101)	$\mathcal{I}+\mathcal{S}$	64.3	65.3
FickleNet [49]	VGG-16	DeepLab (ResNet-101)	$\mathcal{I}+\mathcal{S}$	64.9	65.3
OAA+ [51]	VGG-16	DeepLab (ResNet-101)	$\mathcal{I}+\mathcal{S}$	65.6	66.4
MIL-FCN [44]	VGG-16	-	$\mathcal{I}$	-	25.7
EM-Adapt [46]	VGG-16	-	$\mathcal{I}$	38.2	39.6
SEC [41]	VGG-16	DeepLab (VGG-16)	$\mathcal{I}$	50.7	51.7
MEFF [79]	VGG-16	FCN (VGG-16)	$\mathcal{I}$	-	55.6
RRM [18]	WideResNet-38	DeepLab (VGG-16)	$\mathcal{I}$	60.7	61.0
Araslanov and Roth [95]	WideResNet-38	-	$\mathcal{I}$	62.7	64.3
SSDD [40]	WideResNet-38	WideResNet-38	$\mathcal{I}$	64.9	65.5
RRM [18]	WideResNet-38	DeepLab (ResNet-101)	$\mathcal{I}$	**66.3**	**66.5**
PSA [16] (baseline)	WideResNet-38	DeepLab (VGG-16)	$\mathcal{I}$	58.4	60.5
PSA [16] (baseline)	WideResNet-38	WideResNet-38	$\mathcal{I}$	61.7	63.7
PSA [16]† (baseline)	WideResNet-38	DeepLab (Xception-65)	$\mathcal{I}$	60.7	-
EADER (Ours)	WideResNet-38	DeepLab (ResNet-101)	$\mathcal{I}$	62.5	63.0
EADER (Ours)	WideResNet-38	DeepLab (Xception-65)	$\mathcal{I}$	62.8	63.8