[Re] Background-Aware Pooling and Noise-Aware Loss for Weakly-Supervised Semantic Segmentation

Aryan Mehta, Karan Uppal, Kaushal Jadhav, Monish Natarajan, Mradul Agrawal, Debashish Chakravarty

Indian Institute of Technology Kharagpur

Introduction

- How to efficiently train Segmentation Networks?

  Weakly-Supervised Segmentation

- How to generate better pseudo-labels?

  Background-Aware Pooling

- *How to lessen effect of Noisy Labels?*

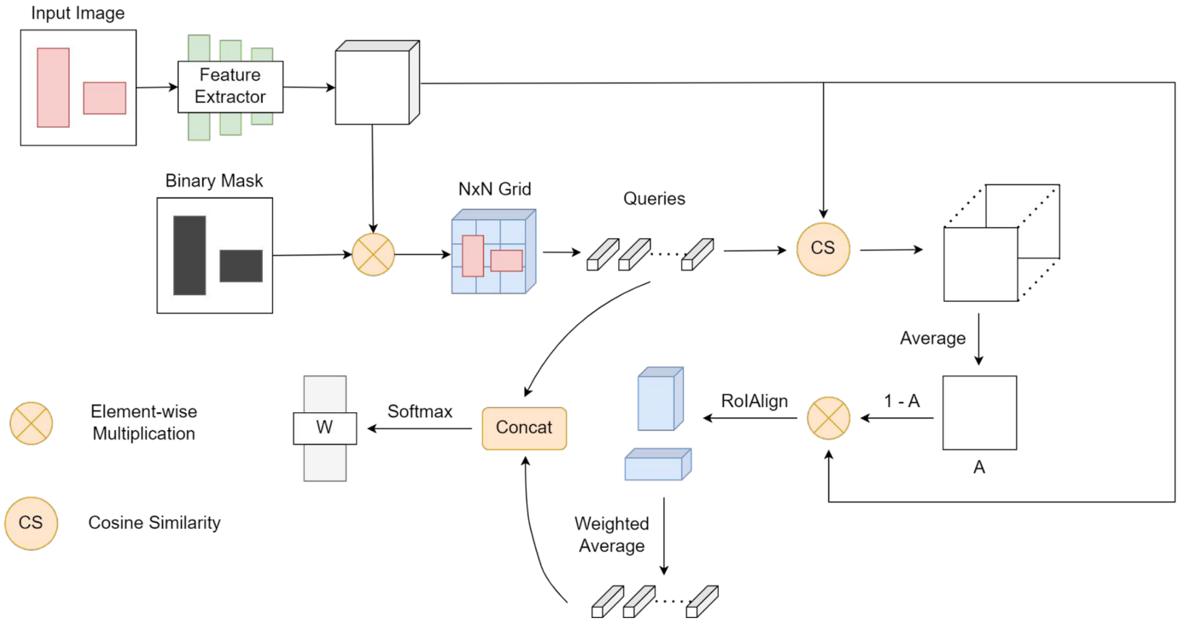
  Noise-Aware Loss

Contributions

- Conducted experiments and verified results of the original paper by implementing in PyTorch Lightning, achieving state-of-the-art results on the Pascal VOC 2012 dataset
- Performed cross dataset evaluation from Pascal VOC 2012 dataset to the MS COCO 2017 dataset to verify the model to be a class agnostic pseudo label generator
- Implemented Noise Aware Loss from scratch and evaluated its performance with other counterpart losses
- Further experiments were conducted to analyze the choice of hyperparameters

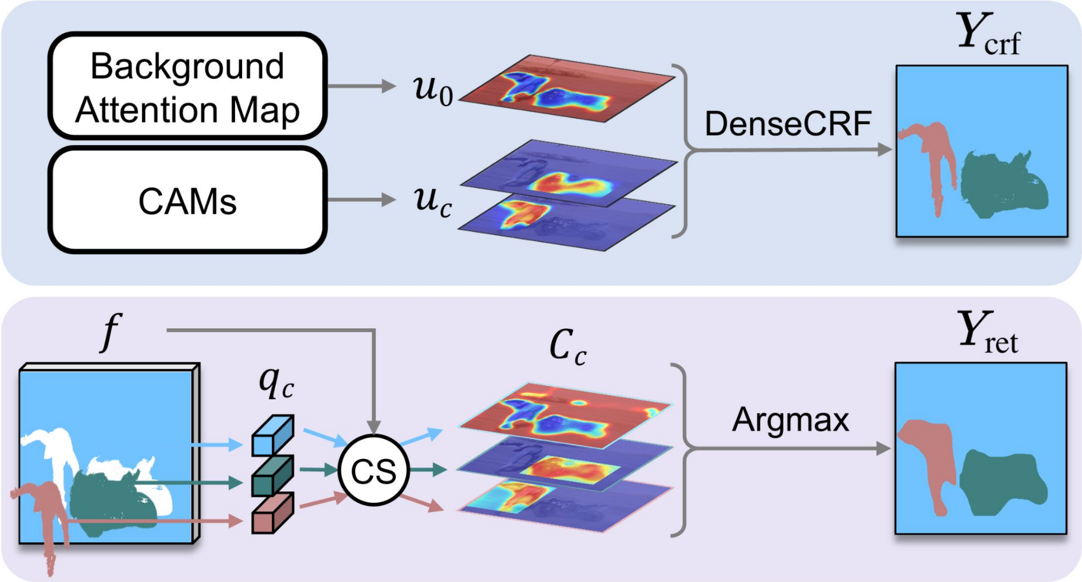
BAP : Background Average Pooling

Assumption: background regions are perceptually consistent in part within an image



BAP computes an attention map for a background adaptively for each image

Pseudo Label generation



We leverage attention maps and CAMs, together with prototypical features, to generate pseudo ground-truth labels.

NAL : Noise Aware Loss

(L+1) dimensional
Probability Map

$$\mathcal{L} = \mathcal{L}_{\text{ce}} + \lambda \mathcal{L}_{\text{wce}}$$

Confidence Map

$$\mathcal{L}_{\text{ce}} = -\frac{1}{\sum_c |\mathcal{S}_c|} \sum_c \sum_{\mathbf{p} \in \mathcal{S}_c} \log H_c(\mathbf{p}) \qquad \mathcal{L}_{\text{wce}} = -\frac{1}{\sum_c \sum_{\mathbf{p} \in \sim \mathcal{S}_c} \sigma(\mathbf{p})} \sum_c \sum_{\mathbf{p} \in \sim \mathcal{S}_c} \sigma(\mathbf{p}) \log H_c(\mathbf{p})$$

S is the set of locations where both Y_{crf} and Y_{ret} have the same label

We exploit a confidence map, using the distances between CNN features for prediction and classifier weights for semantic segmentation, to compute a cross-entropy loss adaptively

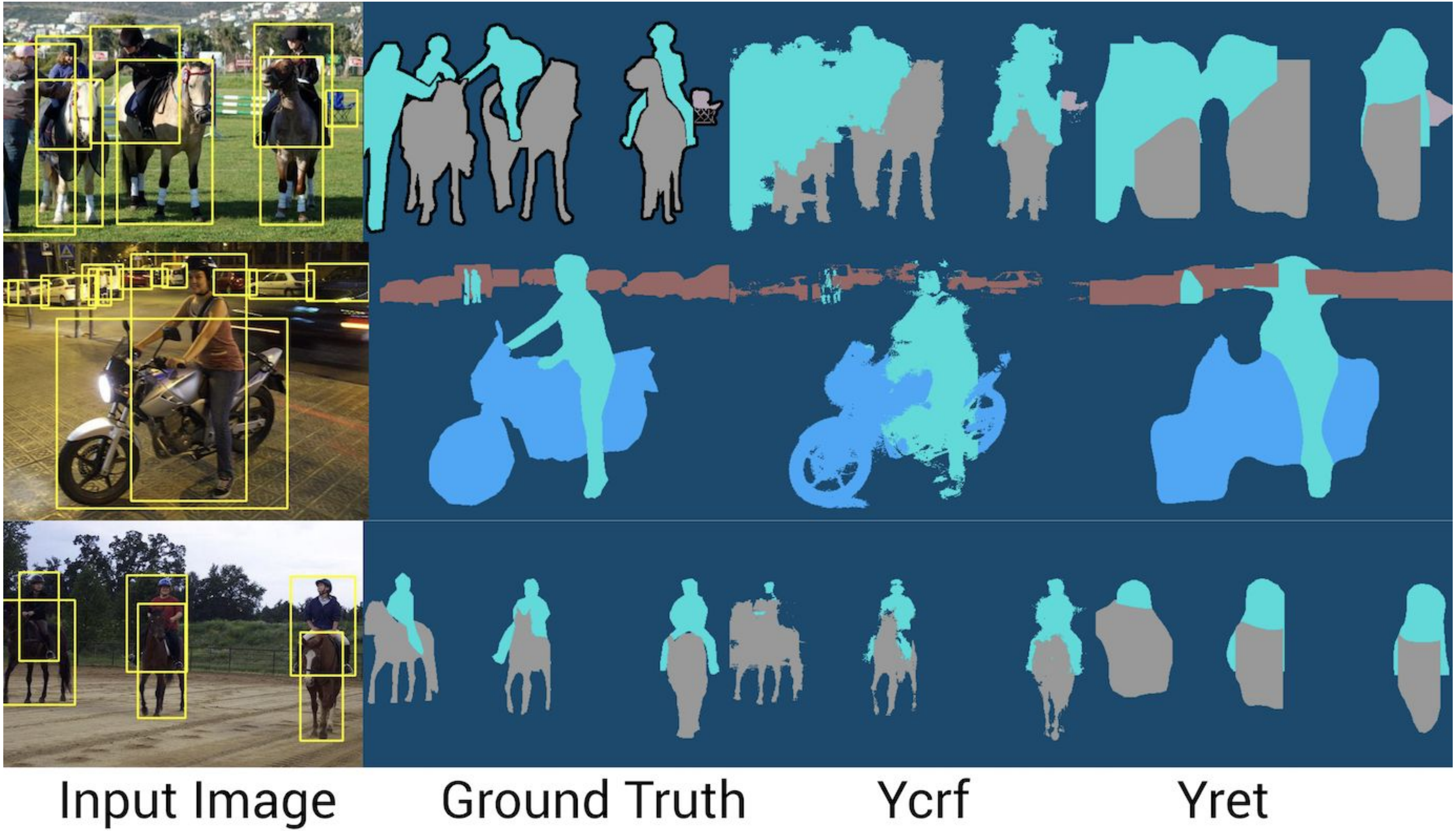
Results: Pseudo Label Generation

Method	Authors' Results	Our Results
GAP	76.1	75.5
BAP Y_{crf} w/o u_0	77.8	77
BAP Y_{crf}	79.2	78.8
BAP Y_{ret}	69.9	69.9
BAP Y_{crf} & Y_{ret}	68.2	72.7

Comparison of IoU scores on Pascal VOC validation pseudo-labels

Method / Results	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
BAP: Y_{crf} $(Authors)$	11.7	28.7	8.0	3.0	15.0	27.1
BAP: Y_{crf} $(Ours)$	8.6	20.1	6.5	1.9	8.8	15.9
BAP: Y_{ret} $(Authors)$	9.0	30.1	2.8	4.4	10.2	16.2
BAP: Y_{ret} $(Ours)$	6.6	20.2	2.5	3.3	5.7	10.6

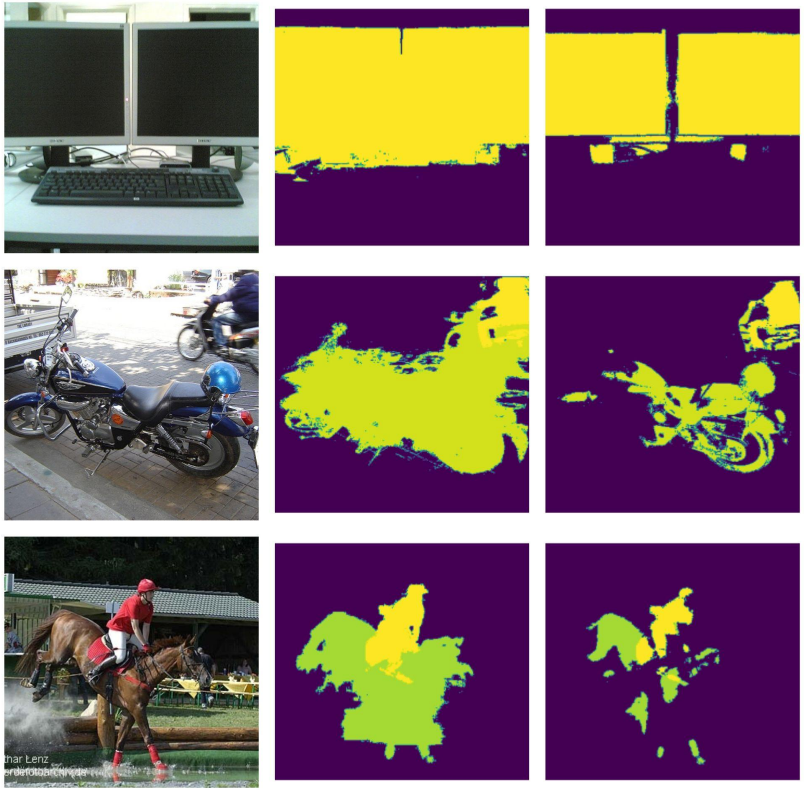
Comparison of mAP scores of pseudo labels on the MS COCO train set for model trained on Pascal VOC


Input Image
Ground Truth
Ycrf
Yret

Results: Class Agnostic Pseudo Label Generator

We exploit the class agnostic foreground attention map for all classes instead of using CAMs and compare its performance.

Method	CAMS for u_c	$1 - u_0$ in place of u_c
BAP Ycrf	78.7	67.48
BAP Yret	70.8	68.66



Original Using CAMs No CAMs

Results: Noise Aware Loss and its counterparts

Method	DeepLab v1		DeepLab v2	
	Author's Results	Our Results	Author's Results	Our Results
w / Y_{crf} (val)	67.8	64.7	74.0	67.0
w / Y_{ret} (val)	66.1	62.8	72.4	70.2
w / NAL (val)	68.1	64.8	74.6	70.8
w / NAL $(test)$	69.4	65.6	76.1	71.7

Results on the Pascal VOC dataset

Method	Authors' Results	Our Results
Baseline	61.8 / 67.5	60.9 / 64.5
w / Entropy Regularization [5]	61.4 / 67.3	60.8 / 64.1
w / Bootstrapping [14]	61.9 / 67.6	60.9 / 64.6
w / $\mathcal{L}_{wce}$	62.4 / 68.1	61.4 / 64.8

Comparison across different losses on the Pascal VOC dataset

Conclusion

- How to generate better pseudo-labels?

  Background-Aware Pooling is a better method for generating pseudo labels as compared to Global Average Pooling

- *How to lessen effect of Noisy Labels?*

  Noise-Aware Loss improves model performance by penalizing incorrect labels adaptively