Regularized Loss for Weakly Supervised Single Class Semantic Segmentation: Supplementary

Olga Veksler

University of Waterloo, Waterloo ON N2L3G1, Canada, oveksler@uwaterloo.ca
https://cs.uwaterloo.ca/ oveksler/

Abstract. The supplementary materials provide additional analysis and experiments on the same datasets and using the same training approach as in the main part of the paper. First, we experimentally evaluate properties of CNN trained with regularized loss. Next, we perform several other test regarding different aspects of training and loss function.

1 Introduction

The supplementary materials are organized as follows. In Sec. 2, we experimentally evaluate the properties of CNN trained with regularized loss. These experiments are directed at understanding whether training with regularized loss just learns to find objects with boundaries overlapping intensity edges, without learning object appearance. This is an important question to ask as edge alignment is the main driving force in our loss function. There are two experiments designed to answer this question. First, in Sec. 2.1 we evaluate how similar are two CNNs with the same architecture but one trained with ground truth, and another with regularized loss. Similarity of the two implies that CNN trained with regularized loss learns object appearance to a degree similar to CNN trained with ground truth. Second, in Sec. 2.2 we train *CNN-reg* on two different datasets and apply them to a third dataset. Different performance on the same dataset implies that *CNN-reg* learns object appearance of the dataset it was trained on.

For all the experiments in the supplementary materials, we only consider *UMobV2* architecture. Since we use only one network architecture, we refer to the network trained with regularized loss as *CNN-reg* and network trained with ground truth as *CNN-gt*.

When training with regularized loss on any dataset, we first train on Oxford-Pet, and then use the weights of the resulting network to train on the desired dataset in the normal stage, skipping annealing. This protocol of gives the best results, see paragraph "Replacing Annealing Stage" in Section 4.1 in the main paper.

2 Experimental Analysis of *CNN-reg*

2.1 Similarity of CNN-gt and CNN-reg

It is interesting to investigate if CNN-gt and CNN-reg share common properties, since their training protocols are rather different. It makes sense to compare if the

performance of CNN-reg is not too far behind that of CNN-gt on the dataset used for training. If CNN-reg is far behind CNN-gt, then these networks are obviously different. Thus we choose to examine the case when training is on OxfordPet. The performance of *CNN-reg* and *CNN-gt* differ in F-measures only by 1.45, see Fig. 4 in the main paper.

It could be the case that while *CNN-reg* and *CNN-gt* perform similarly on OxfordPet, they have learned to look for and make their decisions based on different patterns and, therefore, can be considered as very different.

	CNN-reg	*CNN-gt*	*CNN-reg* -dog	*CNN-gt* -dog	*CNN-reg* -cat	*CNN-gt* -cat
airplane	0.17	0.19	0.07	0.02	0.04	0.10
bike	0.13	0.10	0.04	0.02	0.03	0.09
bird	0.74	0.62	0.58	0.23	0.30	0.33
boat	0.13	0.07	0.07	0.02	0.03	0.04
bottle	0.18	0.07	0.08	0.04	0.02	0.02
bus	0.04	0.00	0.00	0.00	0.01	0.01
car	0.26	0.08	0.07	0.01	0.10	0.07
cat	0.82	0.87	0.81	0.64	0.18	0.20
chair	0.04	0.05	0.03	0.03	0.01	0.02
cow	0.73	0.75	0.33	0.04	0.65	0.70
dining table	0.11	0.05	0.06	0.01	0.00	0.01
dog	0.78	0.84	0.26	0.03	0.70	0.76
horse	0.74	0.81	0.33	0.02	0.70	0.71
motorbike	0.36	0.18	0.06	0.01	0.10	0.10
person	0.42	0.17	0.15	0.03	0.30	0.11
potted plant	0.18	0.14	0.10	0.07	0.05	0.04
sheep	0.80	0.76	0.52	0.09	0.54	0.62
sofa	0.06	0.08	0.07	0.05	0.01	0.01
train	0.05	0.03	0.02	0.00	0.02	0.01
tv	0.02	0.01	0.01	0.00	0.02	0.00

Fig. 1. Experiments on PascalSingle dataset in terms of Jaccard index. Second and third column: results when trained on OxfordPet; forth and fifth columns: results when trained on cat images from OxfordPet with dog images as negative examples; last two columns: results when trained on dog images from OxfordPet with cat images as negative examples

A simple way to test similarity of two trained networks is compare their performance on different datasets without any additional training for those datasets. To make it more informative we need datasets that have objects both similar to and different from those in OxfordPet. We take PASCAL VOC2012 semantic segmentation [1] which has 20 object classes. We select only those images that contain one object class. Thus we get 20 different single object class datasets, and we call them PascalSingle dataset. All images were scaled to size 128×128.

The results of applying *CNN-reg* and *CNN-gt*, without any additional training, on PascalSingle dataset are in the second and third column of Table 2. The performance metric is Jaccard index. The correlation coefficient between their performance is 0.97, showing high similarity. In particular, if we take five classes for which *CNN-reg* performs the best, and five classes for which *CNN-gt* performs the best, they form the same set: $\{bird, cat, dog, horse, cow, sheep\}$, all the animal classes in PascalSingle. This shows that both networks learn animal specific appearance when trained on OxfordPet.

If we want a network to distinguish between the cat and dog class, both in case of training with ground truth and regularized loss, we must provide 'negative' examples. Thus our next experiment is to train on cat images and use dog images as negative examples using negative loss in Eq.(8). The results are in columns four and five of Table 2, for training with regularized loss and ground truth, respectively. As expected, the Jaccard index for the *dog* class drops significantly, both for *CNN-reg* and *CNN-gt*. Performance for the *cat* class stays almost the same for *CNN-reg* but drops by almost 20 for *CNN-gt*. Interestingly, Jaccard index for other non-cat animal classes drops as well, even though these other animal classes were not included in the negative class examples. Out of all animals, the F-measure drops the least for the bird class for both regularized loss and ground truth training versions. Correlation between performance of *CNN-reg* and *CNN-gt* is now 0.8192, reduced, but still significant.

Similarly, we train on OxfordPet dog images using cat images as negative class. The results are in the last two columns in Table 2. Interestingly the cat class get suppressed by a huge margin, but the horse, cow, and sheep classes do not get strikingly suppressed. This holds both for training with ground truth and regularized loss. Correlation between performance of *CNN-reg* and *CNN-gt* is now 0.98, even higher than compared to correlation with no negative examples.

2.2 *CNN-reg* Trained on Different Datasets

In this section, we train *CNN-reg* on two different datasets: OxfordPet [4] and MSRA-B [3], and then we test these two different networks on PascalSingle dataset, separately for each of the 20 classes. The performance, in terms of Jaccard index, is in Fig.2. The second and third columns give Jaccard coefficient when trained on OxfordPet and MSRA, respectively. All images were scaled to size 128×128.

OxfordPet dataset consists of images of dogs and cats. When *CNN-reg* is trained on OxfordPet, it performs well for all the animal classes (bird, cat, cow, dog, horse, sheep), and much worse for all the other classes. MSRA is a saliency dataset. When trained on MSRA, *CNN-reg* performs well for a subset of animal and non-animal classes, likely those classes that tend to be more salient in training images in PascalSingle dataset. Notice that *CNN-reg* trained on MSRA performs significantly worse for most animal classes, compared to *CNN-reg* trained on OxfordPet. Since we are testing both classifiers on the same dataset, this indicates that *CNN-reg* trained on OxfordPet learns animal specific appearance, while *CNN-reg* trained on MSRA learns to segment salient objects. In other

	OxfordPet	MSRA
airplane	0.27	0.56
bike	0.14	0.48
bird	0.71	0.70
boat	0.15	0.44
bottle	0.33	0.52
bus	0.05	0.62
car	0.19	0.63
cat	0.80	0.74
chair	0.11	0.38
cow	0.76	0.74
dining table	0.15	0.28
dog	0.80	0.75
horse	0.74	0.70
motorbike	0.34	0.61
person	0.40	0.54
potted plant	0.19	0.39
sheep	0.77	0.73
sofa	0.11	0.27
train	0.12	0.43
tv	0.02	0.27
mean	0.36	0.54

Fig. 2. Performance (Jaccard index) of *CNN-reg* on PascalSingle dataset when trained on OxfordPet (second column) and MSRA (third column) datasets. Markedly different performance shows that *CNN-reg* learns specifics of object class appearance of the dataset it was trained on.

words, *CNN-reg* learns the appearance of objects in the dataset it was trained on, rather than just learning to extract object that align well to image edges.

3 Other Experiments

We tested the performance of relaxations other than absolute linear in Eq. (1) on the OxfordPet dataset with *UMobV2* architecture. The quadratic relaxation is $w_{pq} \cdot (x_p - x_q)^2$ and has F-score of 72.74, and the bi-linear relaxation is $w_{pq} \cdot (1 - x_p)x_q$ and has F-score of 81.14. Both are much worse than the abolute linear relaxation, which has an F-score of 93.98 on this dataset and network combination. Perhaps the landscape with absolute difference relaxation is more amenable to gradient descent.

We also tested dense-CRF loss from [2] instead of sparse-CRF in our complete loss function in Eq. (7). We tried various annealing schedules for the parameters, and after many experiments the best performance we could get on OxfordPet dataset with *UMobV2* had F-score of 72.13. This is far from random but still a poor performance compared to sparse-CRF. It might be due to the difficulty of optimization, or to a lower correlation of dense-CRF with the accuracy of segmentation (Section 3.1), or both.

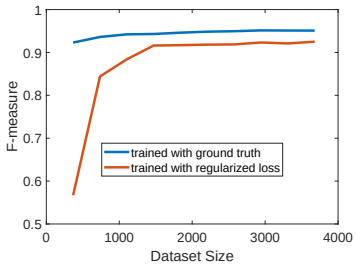

Fig. 3. Dependence of F-measure on the training dataset size. The scatter plots are for *UMobV2* trained on OxfordPet.

We test dependence of segmentation accuracy on size of training dataset. We take *UMobV2* and train it on subsampled OxfordPet dataset. Fig. 3 shows the scatterplot of dataset size vs. F-measure for training with ground truth (in red) and the with regularized loss (in green). Notice that dependence on dataset size when training with regularized loss is much more pronounced. This is reasonable to expect since training without ground truth is a much harder task.

References

1. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision **88**(2), 303–338 (June 2010)
2. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In: Neural Information Processing Systems. pp. 109–117 (2011)
3. Liu, T., Sun, J., Zheng, N.N., Tang, X., Shum, H.Y.: Learning to detect a salient object. In: Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2007)
4. Vedaldi, A.: Cats and dogs. In: Conference on Computer Vision and Pattern Recognition (2012)