Generating Bounding Box Supervision for Semantic Segmentation with Deep Learning

Simone Bonechi, Paolo Andreini(✉), Monica Bianchini, and Franco Scarselli

Department of Information Engineering and Mathematics,
University of Siena, Siena, Italy
paolo.andreini@yahoo.it

Abstract. Most of the leading Convolutional Neural Network (CNN) models for semantic segmentation exploit a large number of pixel–level annotations. Such a human based labeling requires a considerable effort that complicates the creation of large–scale datasets. In this paper, we propose a deep learning approach that uses bounding box annotations to train a semantic segmentation network. Indeed, the bounding box supervision, even though less accurate, is a valuable alternative, effective in reducing the dataset collection costs. The proposed method is based on a two stage training procedure: first, a deep neural network is trained to distinguish the relevant object from the background inside a given bounding box; then, the output of the network is used to provide a weak supervision for a multi–class segmentation CNN. The performances of our approach have been assessed on the Pascal–VOC 2012 segmentation dataset, obtaining competitive results compared to a fully supervised setting.

Keywords: Deep learning · Semantic segmentation
Weak supervision · Bounding box

1 Introduction

Image semantic segmentation is one of the fundamental topic in computer vision. Its goal is to make dense predictions, inferring the label of every pixel within an image. In the last few years, the use of Convolutional Neural Networks (CNNs) has lead to an impressive progress in this field [1–3], yet based on the use of large datasets of fully annotated images. The human annotation procedure for semantic segmentation is particularly expensive, since it requires a pixel–level characterization of images. For this reason, the available datasets are normally orders of magnitude smaller than image classification datasets (f.i. ImageNet [4]). Such a limitation is important, since the performance of CNNs is largely affected by the amount of training examples. On the other hand, bounding box annotations are less accurate than per–pixel annotations, but they are cheaper

L. Pancioni et al. (Eds.): ANNPR 2018, LNAI 11081, pp. 190–200, 2018.
https://doi.org/10.1007/978-3-319-99978-4_15

and easier to be obtained. In this paper, we propose a simple method, called BBSDL – for Bounding Box Supervision with Deep Learning –, to train CNNs for semantic segmentation using only a bounding box supervision (or a mix of bounding box and pixel–level annotations). Figure 1 provides a general overview of our method, that can be sketched as follows.

- A background–foreground network (BF–Net) is trained on a relatively large dataset with a full pixel–level supervision. The aim of the BF–Net is to recognize the most relevant object inside a bounding box.
- A multi–class segmentation CNN is trained on a target dataset, in which the supervision is obtained exploiting the output of the BF–Net.

The rationale behind this approach is that realizing a background–foreground segmentation, constrained to a bounding box, is significantly simpler than obtaining a multi–class semantic segmentation on the whole image. Following this intuition, we consider a scenario in which only bounding box annotations are available on a target dataset. The pixel–level supervision, on such dataset, can be produced from the bounding boxes exploiting the BF–Net trained on a different dataset. In particular, multi–class annotations can be generated in many ways from the output of the BF–Net and, indeed, a set of different solutions were tested, in order to produce the best target for the multi–class segmentation network. The effectiveness of the proposed method has also been compared with other existing techniques [5,6].

The paper is organized as follows. In Sect. 2, we briefly review the state–of–the–art research in semantic segmentation and weakly supervised approaches. Section 3 presents the details of our method, whereas Sect. 4 describes the experimental setup and collects the obtained results. Finally, some conclusions and future perspectives are drawn in Sect. 5.

2 Related Works

Semantic segmentation describes the process of associating each pixel of an image with a class label. Over the past few years, impressive results in image semantic segmentation, so as in many other visual recognition tasks, have been obtained thanks to deep learning techniques [1–3]. Recent semantic segmentation algorithms often convert existing CNN architectures, designed for image classification, to fully convolutional networks. In this framework, semantic segmentation is generally formulated as a pixel–level labeling problem, which requires hand–made fully annotated images. Sadly, producing this kind of supervision is highly demanding and costly. In order to reduce the annotation efforts, some deep learning methods exploit weak supervision. In contrast to learning under strong supervision, these methods are able to learn from weaker annotations, such as image–level tags, partial labels, bounding boxes, etc. In particular, weak supervised learning has been addressed through Multiple Instance Learning (MIL) [7]. MIL deals with training data arranged in sets, called bags, with the supervision provided only at the set level, while single instances are not individually labeled

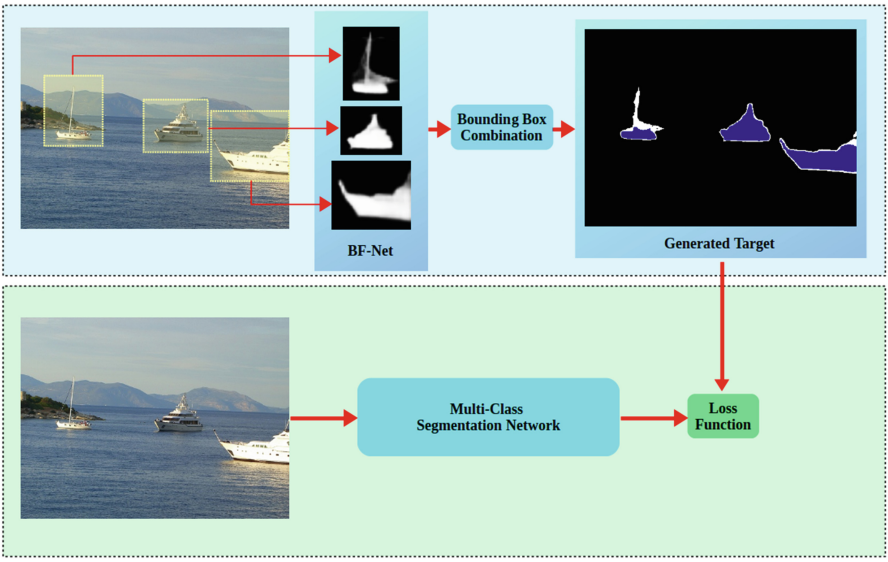


Fig. 1. The training scheme. On the top, weak segmentation annotations are generated from the BF–Net output. At the bottom, the multi–class network is trained based on the generated weak supervision.

[8]. For instance, in [9], a semantic texton forest approach—based on ensembles of decision trees that act directly on image pixels—, revisited in the MIL framework, is proposed for semantic segmentation. Instead, a MIL formulation of multi–class semantic segmentation, by a fully convolutional network, is presented in [10]. MIL extensions to classical segmentation approaches are also introduced in [11] and [12]. Finally, the recently proposed WILDCAT method [13] exploits only global image labels to train deep convolutional neural networks to perform image classification, point–wise object localization, and semantic segmentation.

On the other hand, following an approach which is something similar to our proposal, i.e. that of using bounding box labeling to aid semantic segmentation, in [5], the BoxSup method is proposed, where the core idea is that to iterate between automatically generating region proposals and training convolutional networks. Similarly, in [6], an Expectation–Maximization algorithm was used to iteratively update the training supervision. Nevertheless, while both the above described methods rely on an iterative procedure, our approach directly produces the segmentation supervision, exploiting a deep convolutional network.

3 The BBSDL Method

In the following, we delve into the details of the multi stage training algorithm proposed in this paper (see Fig. 1).

BF–Net Training. The first step in the proposed approach consists in training a deep neural network, capable of recognizing the most relevant object inside a bounding box, thus separating the background from the foreground, called BF–Net (top of Fig. 1). Our experiments are conducted using the Pyramid Scene Parsing architecture [3] (PSP, see Fig. 2).

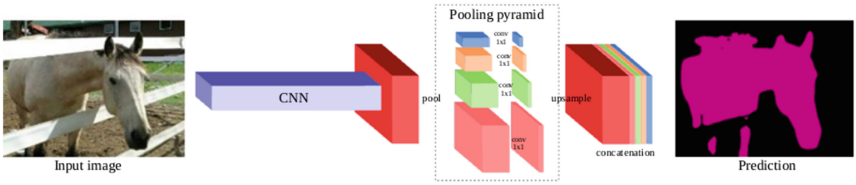


Fig. 2. Scheme of the pyramid scene parsing network, proposed by [3], used in this paper.

The PSP net is a deep fully convolutional neural network which re–purposes the ResNet [14], originally designed for image classification, to perform semantic segmentation. Differently from the original ResNet, a set of dilated convolutions [2] replaces standard convolutions to enlarge the receptive field of the neural network. To gather context information, the PSP exploits a pyramid of pooling with different kernel size. Both upsampling and concatenation produce the final feature representation, which is fed into a convolutional layer to get the desired per–pixel predictions. ResNets of different depths (i.e. with a different number of convolutional layers) were proposed in the original paper [14]. We chose to use the ResNet50 architecture, due to computational issues.

To train this network a dataset composed by image crops is required. Each crop should contain only a single relevant object, in which pixels are annotated either as foreground or background; the information about the object class is not needed and indeed it is not used during training. We employ the COCO dataset [15], which collects instance–level fully annotated images (i.e., in which objects of the same category are labeled separately). Such supervision can be used to extract the bounding box that encloses each object and its background–foreground pixel–wise annotation. The images are then cropped, using the obtained bounding boxes. Moreover, in order to include more context information, each crop is enlarged by 5%, compared with the corresponding box dimensions. Image crops are finally used for training and validating the BF–Net.

Multiclass Dataset Generation. Once the BF–Net has been trained, the pixel–level supervision for the multi–class segmentation network training is generated (bottom of Fig. 1). All the bounding box annotations in the target dataset need to be replaced with a multi–class pixel–level supervision. To this aim, the BF–Net is used to produce predictions over each bounding box. Different strategies can be employed in order to convert such predictions into the final segmentation supervision. In particular, if the naïve approach consists in directly

replacing each bounding box with the pixel–level classification given by the BF–Net, a more refined strategy suggests to use the value of the BF–Net output probability $prob(x, y)$, at position (x, y), to obtain the label $l(x, y)$ for the same point:

$$l(x,y) = \begin{cases} background & \text{if} \quad prob(x,y) < th_1 \\ foreground & \text{if} \quad prob(x,y) > th_2 \\ uncertain & \text{otherwise} \end{cases} \tag{1}$$

The thresholds th_1 and th_2, after a trial–and–error procedure, have been fixed to 0.3 and 0.7, respectively. If $prob(x, y) \in (th_1, th_2)$, then (x, y) is labeled as uncertain and will not be considered for the gradient computation.

Based on both these strategies, a problem naturally arises when bounding box annotations partially overlap. Indeed, in this situation, it is not clear which prediction should be trusted. To solve the ambiguity, three different heuristic approaches were used in the experiments, which are sketched in the following.

- **Ignore Intersection** – Overlapping regions are labeled as "uncertain", so that the gradient will not be propagated in these regions.
- **Smallest Box** – Overlapping regions are considered to belong to the smallest bounding box, which is supposed to coincide with the foreground object.
- **Fixed Threshold** – Overlapping regions are considered to belong to the bounding box with the highest foreground probability prediction.

In Sect. 4.2, we review the experimental results obtained using the three different strategies.

Multiclass Segmentation Network Training. Once the pixel–level supervision is provided, the multi–class network can be trained. In all the experiments, the Pascal–VOC 2012 dataset [16] has been exploited for the PSP training and validation. Similarly to the BF–Net, we used the PSP50 as the multi–class segmentation network, with 21 probability output maps. The experimental details are reported in Sect. 4.3.

Implementation Details. Both the BF–Net and the multi–class segmentation network are implemented in TensorFlow. All the experiments follow the same training procedure that will be explained in the following. Actually, the training phase is composed of two different stages. First, the images are resized at a fixed resolution of 233×233, using padding to maintain the original aspect ratio; early stopping is implemented based on the validation set. Then, the training continues using random crops of 233×233 pixels to obtain a more accurate prediction. The Adam optimizer [17], with learning rate set to 10^{-6} and a mini–batch of 15 examples, has been used to train the network. The evaluation phase relies on a sliding window approach. The experimentation was carried out in a Debian environment, with a single NVIDIA GeForce GTX 1080 Ti GPU, with 128 GB of RAM. The average inference time for each image is about 1.6 s and depends on its size.

4 Experiments and Results

In Sect. 4.1, we describe the datasets used in our experiments, whereas the weak supervision generation is presented in Sect. 4.2. Finally the experimental results are discussed in Sect. 4.3.

4.1 The Datasets

COCO–2017. The COCO–2017 dataset [15], firstly released by Microsoft Corporation, collects 115000 training and 5000 validation instance–level fully annotated images. Also a test set of 41000 images is provided. The object categories are 80, plus the background. However in our experiments, the class supervision is not used. From the given annotations, 816590 and 34938 bounding boxes have been extracted, respectively, for training and evaluating the BF–Net.

Pascal–VOC 2012. The original Pascal–VOC 2012 segmentation dataset collects 1464 training and 1449 validation pixel–level fully annotated images. A test set of 1456 images is also provided, yet without a publicly available labeling. The object categories are 20, plus the background class and a "don't care" class, to account for uncertain regions. Finally, a set of 14212 additional images are provided, with only bounding box annotations. Following the procedure reported in [18], an augmented Pascal–VOC segmentation set was also devised, which provides full pixel–level annotations for 9118 out of the 14212 images originally weakly annotated, yielding a total of 10582 training images. The Pascal–VOC dataset is used for training and evaluating the multi–class segmentation network.

4.2 Weak Supervision Generation for Pascal–VOC 2012

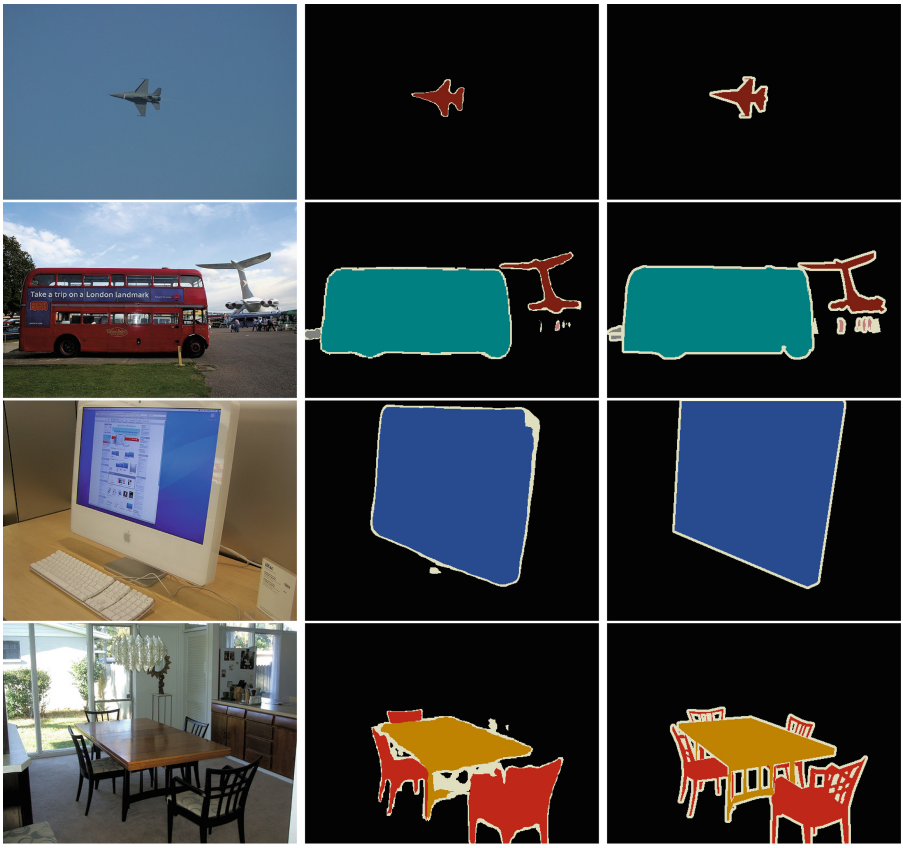
The generation of weak supervisions for the Pascal–VOC dataset follows the procedure described in Sect. 3. First, the BF–Net is trained on the COCO dataset. All the bounding box annotations of the 10582 augmented Pascal–VOC images are then replaced with the multi–class pixel–level supervision obtained from the output of the BF–Net.

Table 1 compares the generated weak supervisions with the strong annotations provided by the Pascal–VOC dataset. Based on the reported results, the best performances are obtained using the "Fixed Threshold" approach, providing an improvement of more than 4% of the mean Intersection over Union (mean IoU)[1], compared to the other methods. It is also worth noting that, when the probability falls between the two thresholds, an uncertainty region is produced. This region, as depicted in Fig. 3, mostly coincides with the uncertainty class present in the Pascal–VOC annotations.

[1] The Mean Intersection over Union is a common measure used to evaluate the quality of a segmentation algorithm, and adopted by the Pascal–VOC competitions. The mean IoU is defined as the average of the ratios $|T \cap P|/|T \cup P|$ for all the images in the test set, where P is the set of pixels predicted as foreground, T is the set of pixels actually annotated as foreground, and $|\cdot|$ denotes the set cardinality operator.

Table 1. Comparison between the Pascal–VOC annotations and the annotations generated by BBSDL.

Supervision generation approach	Mean IoU
Ignore intersection	76.22%
Smallest box	78.01%
Fixed threshold	82.32%



(a) (b) (c)

Fig. 3. Qualitative comparison between Ground–Truth segmentation and generated annotations. (a) Original image. (b) Generated annotations with a fixed threshold. (c) Ground–Truth segmentation.

Table 2 reports the results obtained by training the multi–class segmentation network on the Pascal–VOC validation set, confirming the Fixed Threshold approach as the most effective. For this reason, this setup will be used in all the following experiments.

Table 2. Results of the multi–class segmentation on the Pascal–VOC validation set, obtained with different strategies, for the pixel–level weak supervision generation.

Supervision generation approach	Mean IoU
Ignore intersection	60.93%
Smallest box	60.64%
Fixed threshold	65.28%

4.3 Experimental Results

In order to evaluate the proposed framework, we set up the following experiments, that simulate a different availability of pixel–level and bounding box annotations.

- **Mask supervised setting** – This is the baseline method, in which all the 10582 pixel–level annotations of the Pascal–VOC training set are used.
- **BoundingBox supervised setting** – The pixel–wise labeling provided by the Pascal–VOC dataset is totally disregarded. All the bounding boxes are replaced with the supervision provided by the BF–Net.
- **Semi supervised setting** – This simulate the situation in which a relatively reduced number of pixel–wise annotations is available, whereas it is possible to rely on a greater set of bounding box annotations. As in [5] and [6], we used 1464 strongly supervised pixel–level annotations, replacing the bounding boxes in the remaining 9118 images with the supervision provided by BBSDL.

Table 3 shows the results obtained by BBSDL on the Pascal–VOC 2012 validation set, with the three different experimental setups, compared with other state–of–the–art methods, namely BoxSup [5] and Box–EM [6]. A qualitative evaluation is reported in Fig. 4.

Training with strong annotations produces the best mean IoU on the validation set (70.41%). Instead, the mean IoU drops to 65.28%, using only weak bounding box annotations. On the other hand, the semi–supervised setup allows to obtain a mean IoU of 69.20%[2], which is just 1.21 point worse than the strongly supervised setup. As expected, the performance achieved by using only bounding box annotations is less than that obtainable with a strong supervision. However, the produced results show that BBSDL is viable to be used in practical applications, where strong annotations are not available or, in general, are too expensive to be produced. On the validation set, the difference in performance

[2] We report the results of the semi–supervised approach just for the sake of completeness, since the real purpose of this paper is to present a method that can work on a dataset where no strong supervision is available.

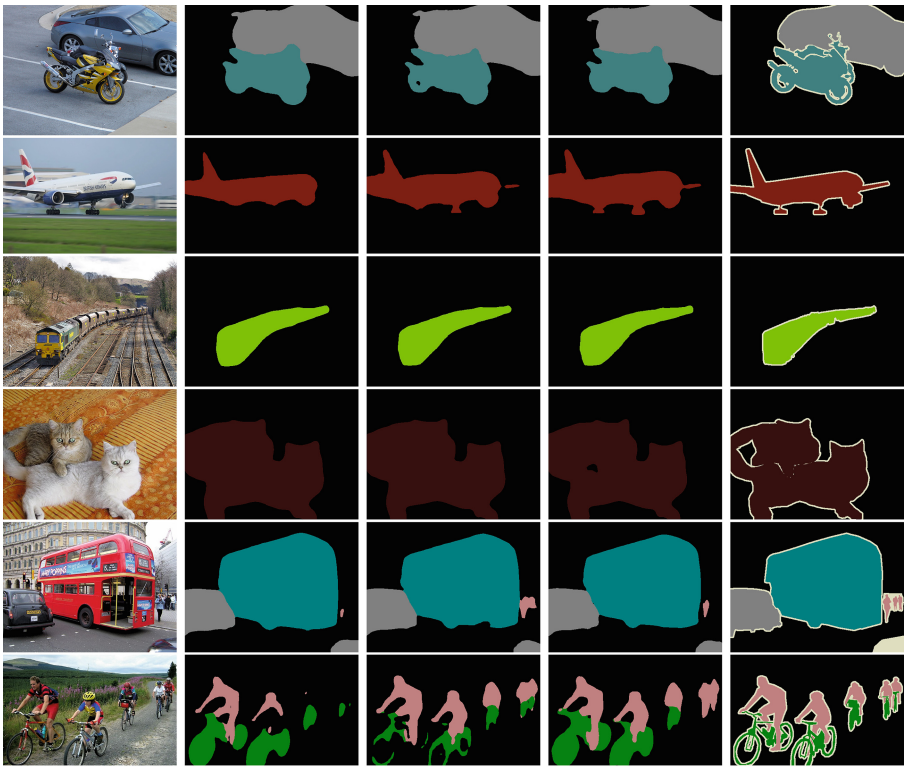


Fig. 4. Qualitative comparison of the results obtained with the three different supervision strategies. (a) Original image. Segmentation obtained by weak bounding box annotations (b), based on the semi–supervised setting (c), and by pixel–level annotations (d). (e) Ground–Truth segmentation.

of BBSDL compared to the strong–supervised (Mask) and the weakly supervised (BoundingBox) cases is 5.13%. This result outperforms that obtained by the Box–EM approach (with a difference of 7%), but it is worse with respect to BoxSup. However, BoxSup employs the MCG segmentation proposal mechanism [19], previously trained on the pixel–level annotations of the Pascal–VOC training set. In Table 4, the results on the Pascal–VOC test set are reported, which look similar to those obtained on the validation set. Unfortunately, the baseline results for BoxSup on the test set are not reported in [5], whereas Box–EM uses a different number of training images—differently from BBSDL and BoxSup, Box–EM also uses the validation images to train the model. For this reason, the comparative evaluation is possible only on the validation set.

Table 3. Comparative results on the Pascal–VOC 2012 validation set.

Method	Supervision	Num. of strong tag	Num. of weak tag	Mean IoU
BoxSup [5]	Mask	10582	-	63.80%
BoxSup [5]	BoundingBox	-	10582	62.00%
BoxSup [5]	Semi	1464	9118	63.50%
Bbox–EM [6]	Mask	10582	-	67.60%
Bbox–EM [6]	BoundingBox	-	10582	60.60%
Bbox–EM [6]	Semi	1464	9118	65.10%
BBSDL	Mask	10582	-	70.41%
BBSDL	BoundingBox	-	10582	65.28%
BBSDL	Semi	1464	9118	69.20%

Table 4. Comparative results on the Pascal–VOC 2012 test set.

Method	Supervision	Num. of strong tag	Num. of weak tag	Mean IoU
BoxSup [5]	Mask	10582	-	-
BoxSup [5]	BoundingBox	-	10582	64.4%
BoxSup [5]	Semi	1464	9118	66.2%
Bbox–EM [6]	Mask	12031	-	70.3%
Bbox–EM [6]	BoundingBox	-	12031	62.2%
Bbox–EM [6]	Semi	1464	10567	66.6%
BBSDL	Mask	10582	-	70.36%
BBSDL	BoundingBox	-	10582	66.24%
BBSDL	Semi	1464	9118	70.25%

5 Conclusions and Future Perspectives

This paper explores the use of bounding box annotations for the training of a state–of–the–art semantic segmentation network. The output of a background–foreground network, capable of recognizing the most relevant object inside a region, has been used to deduce pixel–wise annotations. A fixed threshold strategy has been employed in order to convert the background–foreground network output into the final segmentation supervision. Actually, the obtained weak supervision allowed to train a multi–class segmentation network, whose performances are competitive with respect to approaching the semantic segmentation problem in a strongly–supervised framework. In perspective, how to avoid the use of predefined thresholds for the multi–class dataset generation represents an important issue to deal with, in order to improve the BBSDL performances. Moreover, also training the BF–Net based on unsupervised data should be a matter of future work, capturing images from videos and exploiting the temporal information related to successive frames.

References

1. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of IEEE CVPR 2015, pp. 3431–3440 (2015)
2. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.: Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: Proceedings of ICLR 2015 (2015)
3. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of IEEE CVPR 2017, pp. 6230–6239 (2017)
4. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: Proceedings of IEEE CVPR 2009, pp. 248–255 (2009)
5. Dai, J., He, K., Sun, J.: Boxsup: exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In: Proceedings of IEEE ICCV 2015, pp. 1635–1643 (2015)
6. Papandreou, G., Chen, L.-C., Murphy, K., Yuille, A.L.: Weakly and semi-supervised learning of a DCNN for semantic image segmentation. In: Proceedings of IEEE ICCV 2015, pp. 1742–1750 (2015)
7. Dietterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artif. Intell. **89**(1–2), 31–71 (1997)
8. Carbonneau, M.-A., Cheplygina, V., Granger, E., Gagnon, G.: Multiple instance learning: a survey of problem characteristics and applications. Pattern Recognit. **77**, 329–353 (2018)
9. Vezhnevets, A., Buhmann, J.M.: Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In: Proceedings of IEEE CVPR 2010, pp. 3249–3256 (2010)
10. Pathak, D., Shelhamer, E., Long, J., Darrell, T.: Fully convolutional multi-class multiple instance learning. In: Proceedings of ICLR 2015 (2015)
11. Pinheiro, P.O., Collobert, R.: From image-level to pixel-level labeling with convolutional networks. In: Proceedings of IEEE CVPR 2015, pp. 1713–1721 (2015)
12. Pathak, D., Krahenbuhl, P., Darrell, T.: Constrained convolutional neural networks for weakly supervised segmentation. In: Proceedings of IEEE CVPR 2015, pp. 1796–1804 (2015)
13. Durand, T., Mordan, T., Thome, N., Cord, M.: Wildcat: weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In: Proceedings of IEEE CVPR 2017, pp. 5957–5966 (2017)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of IEEE CVPR 2016, pp. 770–778 (2016)
15. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, David, Pajdla, Tomas, Schiele, Bernt, Tuytelaars, Tinne (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
16. Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL visual object classes challenge: a retrospective. Int. J. Comput. Vis. **111**(1), 98–136 (2015)
17. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: Proceedings of ICLR 2015 (2015)
18. Hariharan, B., Arbeláez, P., Bourdev, L., Maji, S., Malik, J.: Semantic contours from inverse detectors. In: Proceedings of IEEE ICCV 2011, pp. 991–998 (2011)
19. Arbeláez, P., Pont-Tuset, J., Barron, J.T., Marques, F., Malik, J.: Multiscale combinatorial grouping. In: Proceedings of IEEE CVPR 2014, pp. 328–335 (2014)