Weakly Supervised Object Boundaries

Anna Khoreva[1] Rodrigo Benenson[1] Mohamed Omran[1] Matthias Hein[2] Bernt Schiele[1]

[1]Max Planck Institute for Informatics, Saarbrücken, Germany
[2]Saarland University, Saarbrücken, Germany

Abstract

State-of-the-art learning based boundary detection methods require extensive training data. Since labelling object boundaries is one of the most expensive types of annotations, there is a need to relax the requirement to carefully annotate images to make both the training more affordable and to extend the amount of training data. In this paper we propose a technique to generate weakly supervised annotations and show that bounding box annotations alone suffice to reach high-quality object boundaries without using any object-specific boundary annotations. With the proposed weak supervision techniques we achieve the top performance on the object boundary detection task, outperforming by a large margin the current fully supervised state-of-the-art methods.

1. Introduction

Boundary detection is a classic computer vision problem. It is an enabling ingredient for many vision tasks such as image/video segmentation [1, 12], object proposals [17], object detection [37], and semantic labelling [2]. Rather than image edges, many of these tasks require class specific objects boundaries. These are the external boundaries of object instances belonging to a specific class (or class set).

State-of-the-art boundary detection is obtained via machine learning which requires extensive training data. Yet, instance-wise boundaries are amongst the most expensive types of annotations. Compared to two clicks for a bounding box, delineating an object requires a polygon with 20~100 points, i.e. at least 10× more effort per object.

In order to make the training of new object classes affordable, and/or to increase the size of the models we train, there is a need to relax the requirement of high-quality image annotations. Hence the starting point of this paper is the following question: is it possible to obtain object-specific boundaries without having any object boundary annotations at training time?

In this paper we focus on learning object boundaries in a weakly supervised fashion and show that high quality object boundary detection can be obtained without using any class-specific boundary annotations. We propose several ways of generating object boundary annotations with different levels of supervision, from just using a bounding box oriented object detector to using the boundary detector trained on generic boundaries. For generating weak object boundary annotations we consider different sources, fusing unsupervised image segmentation [11] and object proposal methods [32, 25] with object detectors [14, 27]. We show that bounding box annotations alone suffice to achieve objects boundary estimates with high quality.

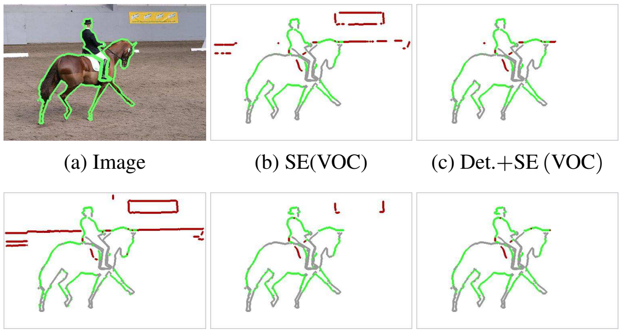

(a) Image (b) SE(VOC) (c) Det.+SE (VOC)
(d) SE(BSDS) (e) SE (weak) (f) Det.+SE (weak)

Figure 1: Object-specific boundaries 1a differ from generic boundaries (such as the ones detected in 1d). The proposed weakly supervised approach drives boundary detection towards the objects of interest. Example results in 1e and 1f. Red/green indicate false/true positive pixels, grey is missing recall. All methods shown at 50% recall.

We present results using a decision forest [9] and a convnet edge detector [35]. We report top performance on Pascal object boundary detection [16, 10] with our weak-supervision approaches already surpassing previously reported fully supervised results.

Our main contributions are summarized below:

- We introduce the problem of weakly supervised object-specific boundary detection.
- We show that good performance can be obtained on BSDS, PascalVOC12, and SBD boundary estimation using only weak-supervision (leveraging bounding box detection annotations without the need of instance-wise object boundary annotations).
- We report best known results on PascalVOC12, and SBD datasets. Our weakly supervised results alone improve over the previous fully supervised state-of-the-art.

The rest of this paper is organized as follows. Section 3 describes different types of boundary detection and the considered datasets. In Section 4 we investigate the robustness to annotation noise during training. We leverage our findings and propose several approaches for generating weak boundary annotations in Section 5. Sections 6-9 report results using the two different classifier architectures.

2. Related work

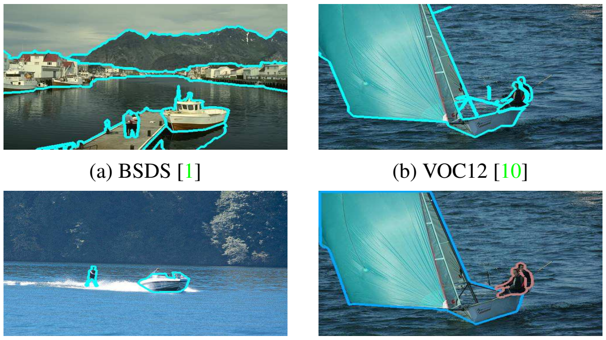
Generic boundaries Boundary detection has been regained attention recently. Early methods are based on a fixed prior model of what constitutes a boundary (e.g. Canny [6]). Modern methods leverage machine learning to push performance. From well crafted features and simple classifiers (gPb [1]), to powerful decision trees over fixed features (SE [9], OEF [15]), and recently to end-to-end learning via convnets (DeepEdge [3], N4 [13], HFL [4], BNF [5], HED [35]). Convnets are usually pre-trained on large classification datasets, so as to be initialized with reasonable features. The more sophisticated the model, the more data is needed to learn it.

Other than pure boundary detection, segmentation techniques (such as F&H [11], gPb-owt-ucm [1], and MCG [25]), can also be used to improve or to generate closed contours.

A few works have addressed unsupervised detection of generic boundaries [19, 20]. PMI [19] detects boundaries by modelling them as statistical anomalies amongst all local image patches, reaching competitive performance without the need for training. Recently [20] proposes to train edge detectors using motion boundaries obtained from a large corpus of video data in place of human supervision. Both approaches reach similar detection performance.

Object-specific boundaries In many applications, there is interest to focus on boundaries of specific object classes. The class-specific object boundary detectors need then to be trained or tuned to the classes of interest. This problem is more recent and still relatively unexplored. [16] introduced the SBD dataset to measure this task over the 20 pascal categories. [16] proposes to re-weight generic boundaries using the activation regions of a detector. [31] proposed to train class-specific boundary detectors, and weighted them at test time according to an image classifier. More recently [4, 5] consider mixing a semantic labelling convnet with a generic boundary detection convnet, to obtain class specific boundaries.

Weakly supervised learning In this work we are interested in object-specific boundaries *without using class specific boundary annotation*s. We only use bounding box annotations, and in some experiments, generic boundaries (from BSDS [1]). Multiple works have addressed weakly supervised learning for object localization [23, 7], object detection [26, 34], or semantic labelling [33, 36, 24]. To the best of our knowledge there is no previous work attempting to learn object boundaries in a weakly supervised fashion.



(a) BSDS [1]

(b) VOC12 [10]

(c) COCO [21]

(d) SBD [16]

Figure 2: Datasets considered.

3. Boundary detection tasks

In this work we distinguish three types of boundaries: generic boundaries ("things" and "stuff"), instance-wise boundaries (external object instance boundaries), and class specific boundaries (object instance boundaries of a certain semantic class). For detecting these three types of boundaries we consider different datasets: BSDS500 [1, 22], Pascal VOC12 [10], MS COCO [21], and SBD [16], where each represents boundary annotations of a given boundary type (see Figure 2).

BSDS We first present our results on the Berkeley Segmentation Dataset and Benchmark (BSDS) [1, 22], the most established benchmark for generic boundary detection task. The dataset contains 200 training, 100 validation and 200 test images. Each image has multiple ground truth annotations. For evaluating the quality of estimated boundaries three measures are used: fixed contour threshold (ODS), per-image best threshold (OIS), and average precision (AP). Following the standard approach [9, 6] prior to evaluation we apply a non-maximal suppression technique to boundary probability maps to obtain thinned edges.

VOC For evaluating instance-wise boundaries we propose to use the PASCAL VOC 2012 (VOC) segmentation dataset [10]. The dataset contains 1 464 training and 1 449 validation images, annotated with contours for 20 object classes for all instances. The dataset was originally designed for semantic segmentation. Therefore only object interior pixels are marked and the boundary location is recovered from the segmentation mask. Here we consider only object boundaries without distinguishing the semantics, treating all 20 classes as one. For measuring the quality of predicted boundaries the BSDS evaluation software is used. Following [31] the maxDist (maximum tolerance for edge match) is set to 0.01.

COCO To show generalization of the proposed method for instance-wise boundary detection we use the MS COCO

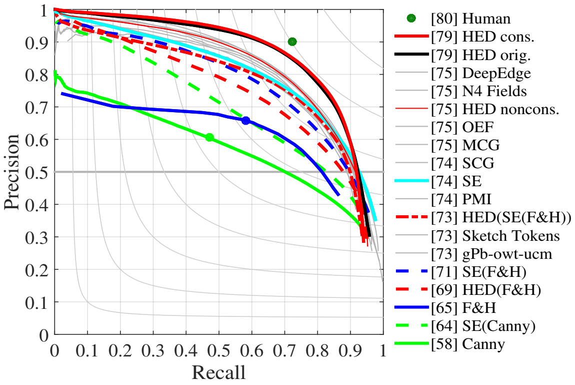


Figure 3: BSDS results. Canny and F&H points indicate the boundaries used as noisy annotations. When trained over noisy annotations, both SE and HED provide a large quality improvement.

Family	Method	ODS	OIS	AP	ΔAP%
Unsupervised	Canny	58	62	55	-
	F&H	64	67	64	-
	PMI	74	77	78	-
Trained on ground truth	gPb-owt-ucm	73	76	73	-
	SE(BSDS)	74	76	79	-
	HED(BSDS) noncons.	75	77	80	-
	HED(BSDS) cons.	79	81	84	-
Trained on unsupervised boundary estimates	SE (Canny)	64	67	64	38
	SE (F&H)	71	74	76	80
	SE (SE (F&H))	72	74	76	80
	SE(PMI)	72	75	77	-
	HED (F&H)	69	72	73	56
	HED (SE (F&H))	73	76	75	69

Table 1: Detailed BSDS results, see Figure 3 and Section 4. Underline indicates ground truth baselines, and bold are our best weakly supervised results. $(\cdot)$ denotes the data used for training. ΔAP% indicates the ratio between the same model trained on ground truth, and the noisy input boundaries. The closer to 100%, the lower the drop due to using noisy inputs instead of ground truth.

(COCO) dataset [21]. The dataset provides semantic segmentation masks for 80 object classes. For our experiments we consider only images that contain the 20 Pascal classes and objects larger than 200 pixels. The subset of COCO that contains Pascal classes consists of 65 813 training and 30 163 validation images. For computational reasons we limit evaluation to 5 000 randomly chosen images of the validation set. The BSDS evaluation software is used (maxDist $= 0.01$). Only object boundaries are evaluated without distinguishing the semantics.

SBD We use the Semantic Boundaries Dataset (SBD) [16] for evaluating class specific object boundaries. The dataset consists of 11 318 images from the trainval set of the PASCAL VOC2011 challenge, divided into 8 498 training and 2 820 test images. This dataset has object instance boundaries with accurate figure/ground masks that are also labeled with one of 20 Pascal VOC classes. The boundary detection accuracy for each class is evaluated using the official evaluation software [16]. During the evaluation process all internal object-specific boundaries are set to zero and the maxDist is set to 0.02. We report the mean ODS F-measure (F), and average precision (AP) across 20 classes.

Note that VOC and SBD datasets have overlap between their train and test sets. When doing experiments across datasets we make sure not to re-use any images included in the test set considered.

Baselines For our experiments we consider two different types of boundary detectors - SE [9] and HED [35] - as baselines.

SE is at the core of multiple related methods (SCG, MCG, OEF). SE [9] builds a "structured decision forest" which is a modified decision forest, where the leaf outputs are local boundary patches (16 × 16 pixels) that are averaged at test time, and the split nodes are built taking into account the local segmentation of the ground truth input patches. It uses binary comparison over hand-crafted edge and self-similarity features as split decisions. By construction this method requires closed contours (i.e. segmentations) as training input. This detector is reasonably fast to train/test and yields good detection quality.

HED [35] is currently the top performing convnet for BSDS boundaries. It builds upon a VGG16 network pre-trained on ImageNet [30], and exploits features from all layers to build its output boundary probability map. By also exploiting the lower layers (which have higher resolution) the output is more detailed, and the fine-tuning is more effective (since all layers are guided directly towards the boundary detection task). To reach top performance, HED is trained using a subset of the annotated BSDS pixels, where all annotators agree [35]. These are so called "consensus" annotations [18], and correspond to sparse ∼15% of all true positives.

4. Robustness to annotation noise

We start by exploring weakly supervised training for generic boundary detection, as considered in BSDS.

Model based approaches such as Canny [6] and F&H [11] are able to provide low quality boundary detections. We notice that correct boundaries tend to have consistent appearance, while erroneous detections are mostly inconsistent. Robust training methods should be able to pick-up the signal in such noisy detections.

SE In Figure 3 and Table 1 we report our results when training a structured decision forest (SE) and a convnet (HED) with noisy boundary annotations. By $(\cdot)$ we denote the data used for training. When training SE using either Canny ("SE (Canny)") or F&H ("SE (F&H)") we observe a notable jump in boundary detection quality. Comparing SE trained with the BSDS ground truth (fully supervised, SE (BSDS)), with the noisy labels from F&H,

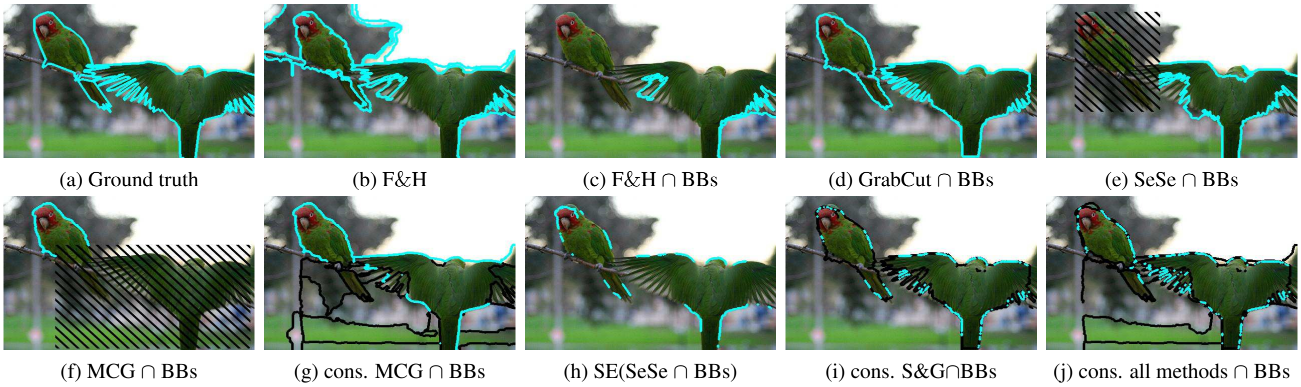

(a) Ground truth (b) F&H (c) F&H ∩ BBs (d) GrabCut ∩ BBs (e) SeSe ∩ BBs (f) MCG ∩ BBs (g) cons. MCG ∩ BBs (h) SE(SeSe ∩ BBs) (i) cons. S&G∩BBs (j) cons. all methods ∩ BBs

Figure 4: Different generated boundary annotations. Cyan/black indicates positive/ignored boundaries.

SE (F&H) closes up to 80% of the gap between SE (F&H) and SE (BSDS) (ΔAP% column in Table 1).
Since the training data of our weak supervision contains label noise (errors), we do not expect results to match the fully supervised case. Still, SE (F&H) is only 3 AP percent points behind from the fully supervised case (76 vs. 79).
We believe that the strong noise robustness of SE can be attributed to the way it builds its leaves. The final output of each leaf is the medoid of all segments reaching it. If the noisy boundaries are randomly spread in the image appearance space, the medoid selection will be robust.

HED The HED convnet [35] reaches top quality when trained over consensus annotations. When using all annotations ("non consensus"), its performance is comparable to other convnet alternatives. When trained over F&H the relative improvement is smaller than for the SE case, when combined with SE (denoted "HED(SE (F&H))") it reaches 69 ΔAP% . HED (SE (F&H)) provides better boundaries than SE (F&H) alone, and reaches quality comparable to the classic gPb method [1] (75 vs. 73).

On BSDS the unsupervised PMI methods provides better boundaries than our weakly supervised variants. However PMI cannot be adapted to provide object-specific boundaries. For this we need to rely on methods than can be trained, such as SE and HED.

Conclusion SE is surprisingly robust to annotation noise during training. HED is also robust but to a lesser degree. By using noisy boundaries generated from unsupervised methods, we can reach a performance comparable to the bulk of current methods.

5. Weakly supervised boundary annotations

Based on the observations in Section 4, we propose to train boundary detectors using data generated from weak annotations. Our weakly supervised models are trained in a regular fashion, but use generated (noisy) training data as input instead of human annotations.

We consider boundary annotations generated with three different levels of supervision: fully unsupervised, using only detection annotations, and using both detection annotations and BSDS boundary annotations (e.g. using generic boundary annotation, but zero object-specific boundaries). In this section we present the different variants of weakly supervised boundary annotations. Some of them are illustrated in Figure 4.

BBs We use the bounding box annotations to train a class-specific object detector [27, 14]. We then apply this detector over the training set (and possibly a larger set of images), and retain boxes with confidence scores above 0.8. We saw no noticeable difference when using directly the ground truth annotations, see supplementary material for details.

F&H As a source of unsupervised boundaries we consider the classical graph based image segmentation technique proposed by [11] (F&H). To focus the training data on the classes of interest, we intersect these boundaries with detection bounding boxes from [27] (**F&H ∩ BBs**). Only the boundaries of segments that are contained inside a bounding box are retained.

GrabCut Boundaries from F&H will trigger on any kind of boundary, including the internal boundaries of objects. A way to exclude internal object boundaries, is to extract object contours via figure-ground segmentation of the detection bounding box. We use **GrabCut** [28] for this purpose. We also experimented with DenseCut [8] and CNN+GraphCut [29], but did not obtain any gain; thus we report only GrabCut results.
For the experiments reported below, for **GrabCut ∩ BBs** a segment is only accepted if a detection from [27] has the intersection-over-union score (IoU) ≥ 0.7. If a detection bounding boxes has no matching segment, the whole region is marked as ignore (see Figure 4e) and not used during the training of boundary detectors.

Object proposals Another way to bias generation of boundary annotations towards object contours is to consider

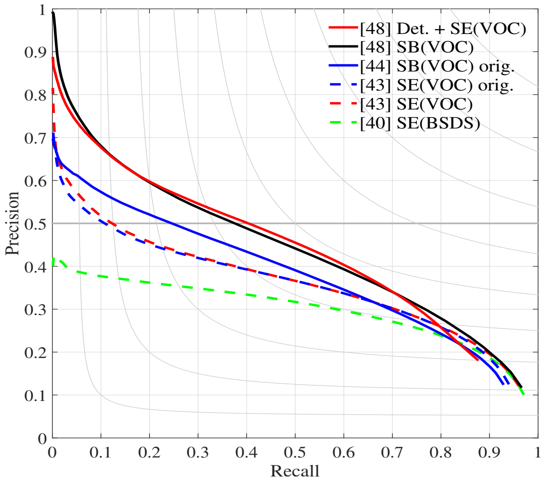


Figure 5: VOC12 results, fully supervised SE models. $(\cdot)$ denotes the data used for training. Continuous/dashed line indicates models using/not using a detector at test time. Legend indicates AP numbers.

Family	Method	Data	Without BBs			With BBs		
			F	AP	ΔAP	F	AP	ΔAP
GT	SE	VOC	43	35	-	48	41	-
Other GT	SE	COCO	44	37	2	49	42	1
	SE	BSDS	40	29	-6	47	39	-2
	MCG		41	28	-7	48	39	-2
Weakly supervised	SE	F&H ∩ BBs	40	29	-6	46	36	-5
		GrabCut ∩ BBs	41	32	-3	47	39	-2
		SeSe ∩ BBs	42	35	0	46	39	-2
		SeSe$_+$ ∩ BBs	**43**	**36**	**+1**	46	39	-2
		MCG ∩ BBs	43	34	-1	47	39	-2
		MCG$_+$ ∩ BBs	43	35	0	**48**	**40**	**-1**
Unsupervised	F&H	-	34	15	-20	41	25	-16
	PMI		41	29	-6	47	38	-3

Table 2: VOC results for SE models, see Figures 5 and 6. Bold indicates our best weakly supervised results.

object proposals. **SeSe** [32] is based on the F&H [11] segmentation (thus it is fully unsupervised), while **MCG** [25] employs boundaries estimated via SE (BSDS) (thus uses generic boundary annotations).

Similar to GrabCut ∩ BBs, **SeSe ∩ BBs** and **MCG ∩ BBs** are generated by matching proposals to bounding boxes (if IoU $\geq$ 0.9). BBs come from [14] with the corresponding object proposals. When more than one proposal is matched to a detection bounding box we use the union of the proposal boundaries as positive annotations. This maximizes the recall of boundaries, and somewhat imitates the multiple human annotators in BSDS. We also experimented using only the highest overlapping proposal, but the union provides marginally better results; thus we report only the latter. Since proposals matching a bounding box might have boundaries outside it, we consider them all since the bounding box itself might not cover well the underlying object.

Consensus boundaries As pointed out in Table 1, HED requires consensus boundaries to reach good performance.

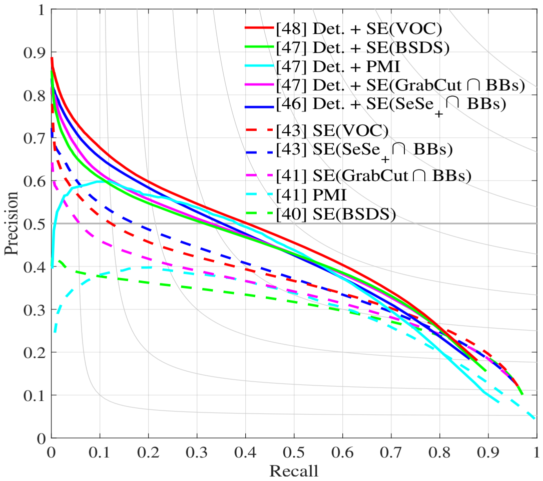


Figure 6: VOC12 results, weakly supervised SE models. $(\cdot)$ denotes the data used for training. Continuous/dashed line indicates models using/not using a detector at test time. Legend indicates AP numbers.

Thus rather than taking the union between proposal boundaries, we consider using the consensus between object proposal boundaries. The boundary is considered to be present if the agreement is higher than 70%, otherwise the boundary is ignored. We denote such generated annotations as "cons.", e.g. **cons. MCG ∩ BBs** (see Figure 4g).

Another way to generate sparse (consensus-like) boundaries, is to threshold the boundary probability map out of SE $(\cdot)$ model. **SE** (**SeSe ∩ BBs**) uses the top 15% quantile per image as weakly supervised annotations.

Finally, other than consensus between proposals, we can also do consensus between methods. **cons. S&G ∩ BBs** is the intersection between SE (SeSe ∩ BBs), SeSe and GrabCut boundaries (fully unsupervised); while **cons. all methods ∩ BBs** is the intersection between MCG, SeSe and GrabCut (uses BSDS data).

Datasets Since we generate boundary annotations in a weakly supervised fashion, we are able to generate boundaries over arbitrary image sets. In our experiments we consider SBD, VOC (segmentation), and VOC$_+$ (VOC plus images from Pascal VOC12 detection task). Methods using VOC$_+$are denoted using $\cdot_+$ (e.g. SE (SeSe$_+$ ∩ BBs)).

6. Structured forest VOC boundary detection

In this section we analyse the variants of weakly supervised methods for object boundary detection proposed in Section 5 as opposed to the fully supervised ones. From now on we are interested in external boundaries of objects. Therefore we employ the Pascal VOC12, treating all 20 Pascal classes as one. See details of the evaluation protocol in Section 3. We start by discussing results using SE; convnet results are presented in Section 7.

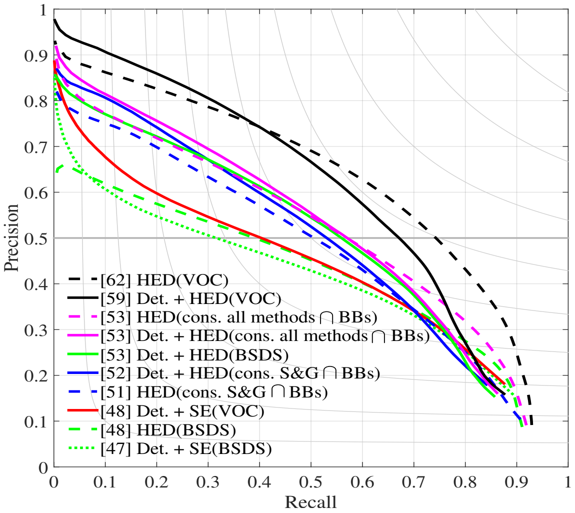


Figure 7: VOC12 HED results. $(\cdot)$ denotes the data used for training. Continuous/dashed line indicates models using/not using a detector at test time. Legend indicates AP numbers.

Family	Method	Data	Without BBs			With BBs		
			F	AP	ΔAP	F	AP	ΔAP
GT	SE	VOC	43	35	-	48	41	-
	HED		62	61	26	59	58	17
Other GT	HED	BSDS	48	41	6	53	48	7
		COCO	59	60	25	56	55	14
Weakly supervised	SE	MCG ∩ BBs	43	34	-1	47	39	-2
	HED	SE(SeSe ∩ BBs)	45	37	3	49	40	-1
		MCG ∩ BBs	50	44	9	48	42	1
		cons. S&G ∩ BBs	**51**	**46**	**+11**	**52**	**47**	**+8**
		cons. MCG ∩ BBs	53	50	15	52	49	8
		cons. all methods∩BBs	**53**	**50**	**+15**	**53**	**50**	**+9**

Table 3: VOC results for HED models, see Figure 7. Bold indicates our best weakly supervised results.

6.1. Training models with ground truth

SE Figure 5 and Table 2 show results of SE trained over the ground truth of different datasets (dashed lines). Our results of SE (VOC) are on par to the ones reported in [31]. The gap between SE (VOC) and SE (BSDS) reflects the difference between generic boundaries and boundaries specific to the 20 VOC object categories (see also Figure 1).

SB To improve object-specific boundary detection, the situational boundary method SB [31], trains 20 class-specific SE models. These models are combined at test time using a convnet image classifier. The original SB results and our re-implementation SB (VOC) are shown in Figure 5. Our version obtains better results (4 percent points gain in AP) due to training the SE models with more samples per image, and using a stronger image classifier [30].

Detector + SE Rather than training and testing with 20 SE models plus an image classifier, we propose to leverage the same training data using a single SE model together with a detector [14]. By computing a per-pixel maximum among all detection bounding boxes and their score, we construct an "objectness map" that we multiply with the boundary probability map from SE. False positive boundaries are thus down-scored, and boundaries in high confidence regions for the detector get boosted. The detector is trained with the same per object boundary annotations used to train the SE model, no additional data is required.

Our Det.+SE (VOC) obtains the same detection quality as SB (VOC) while using only a single SE model. These are the best reported results on this task (top of Table 2), when using the fully supervised training data.

At the cost of more expensive training and test, one could in principle also combine object detection with the situational boundary method [31], this is out of scope of this paper and considered as future work.

6.2. Training models using weak annotations

Given the reference performance of Det.+SE (VOC), can we reach similar boundary detection quality without using the boundary annotations from VOC?

SE $(\cdot)$ First we consider using a SE model alone at test time. Using only the BSDS annotations leads to rather low performance (see SE (BSDS) in Figure 6). PMI shows a similar gap. The same BSDS data can be used to generate MCG object proposals over the VOC training data, and a detector trained on VOC bounding boxes can generate bounding boxes over the same images. We combined them together to generate boundary annotations via MCG ∩ BBs, as described in Section 5. The weak supervision from the bounding boxes can be used to improve the performance of SE (BSDS). By extending the training set to additional pascal images (SE ($\text{MCG}_+ \cap$ BBs) in Table 2) we can reach *the same performance* as when using the VOC ground truth.
We also consider variants that do not leverage the BSDS boundary annotations, such as SeSe and GrabCut. SeSe provides essentially the same result as MCG. Note that both MCG and SeSe are tuned on VOC. Comparing to GrabCut ∩ BBs, a "pascal-agnostic" method, we can see that this bias has a minor impact.

Det.+SE $(\cdot)$ Applying object detection at test time squashes the differences among all weakly supervised methods. Det.+PMI shows strong results, but (since not trained on boundaries) fails to reach high precision. The high quality of Det.+BSDS indicates that BSDS annotations, despite being in principle "generic boundaries" in practice reflect well object boundaries, at least in the proximity of an object. This is further confirmed in Section 7.
Compared to Det.+BSDS our weakly supervised annotation variants further close the gap to Det.+SE (VOC) (especially in high precision area), even when not using any BSDS data.

Conclusion Based only on bonding box annotations, our weakly supervised boundary annotations enable the Det.+

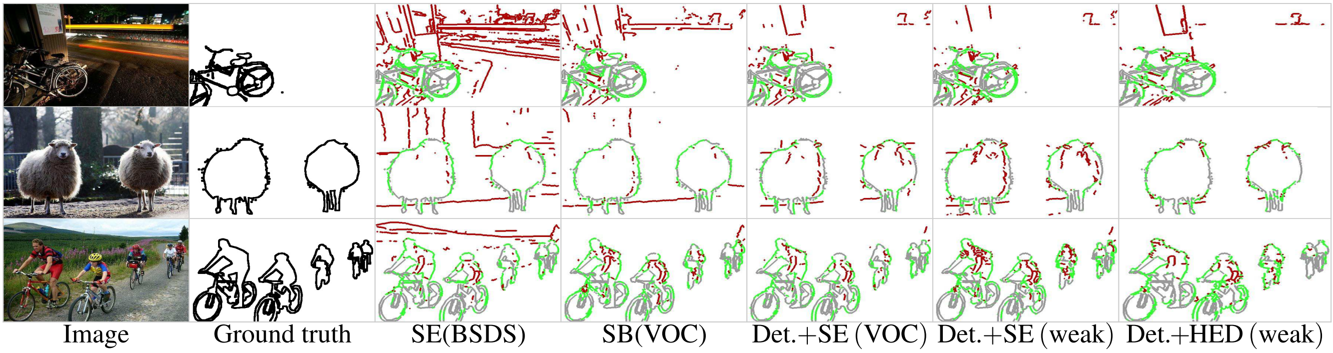


Figure 8: Qualitative results on VOC. $(\cdot)$ denotes the data used for training. Red/green indicate false/true positive pixels, grey is missing recall. All methods are shown at 50% recall. Det.+SE (weak) refers to the model Det.+SE (SeSe$_+ \cap$ BBs) Det.+HED (weak) refers to Det.+HED (cons. S&G $\cap$ BBs). Object-specific boundaries differ from generic boundaries (such as the ones detected by SE(BSDS)). By using an object detector we can suppress non-object boundaries and focus boundary detection on the classes of interest. The proposed weakly supervised techniques allow to achieve high quality boundary estimates that are similar to the ones obtained by fully supervised methods.

Method	Family	Data	Without BBs			With BBs		
			F	AP	ΔAP	F	AP	ΔAP
SE	GT	COCO	40	32	-	45	37	-
	Other GT	BSDS	34	23	-9	43	33	-4
	Weakly supervised	SeSe$_+ \cap$ BBs	**40**	**31**	**-1**	**44**	**35**	**-2**
		MCG$_+ \cap$ BBs	39	30	-2	44	35	-2
HED	GT	COCO	60	59	27	56	55	18
	Other GT	BSDS	44	34	2	49	42	5
	Weakly supervised	cons. S&G$\cap$BBs	47	39	7	48	42	5
		cons. all methods$\cap$BBs	**49**	**43**	**+11**	**50**	**44**	**+7**

Table 4: COCO results, curves in supplementary material. Bold indicates our best weakly supervised results.

SE model to match the fully supervised case, improving over the best reported results on the task. We also observe that BSDS data allows to train models that describe well object boundaries.

7. Convnet VOC boundary detection results

This section analyses the performance of the HED [35] trained with the weakly supervised variants proposed in Section 5. We use our re-implementation of HED which is on par performance with the original (see Figure 3). We use the same evaluation setup as in the previous section. Figure 7 and Table 3 show the results.

HED $(\cdot)$ The HED(VOC) model outperforms the SE(VOC) model by a large margin. We observe in the test images that HED manages to suppress well the internal object boundaries, while SE fails to do so due to its more local nature. Note that HED also leverages the ImageNet pre-training [35].

Even though trained on the generic boundaries HED(BSDS) achieves high performance on the object boundary detection task. HED(BSDS) is trained on the "consensus" annotations and they are closer to object-like boundaries as the fraction of annotators agreeing on the presence of external object boundaries is much higher than for non-object or internal object boundaries.

For training HED, in contrast to SE model, we do not need closed contours and can use the consensus between different weak annotation variants. This results in better performance. Using the consensus between boundaries of MCG proposals HED(cons. MCG $\cap$ BBs) improves AP by 6% compared to using the union of object proposals HED(MCG $\cap$ BBs) (see Table 3) .

The HED models trained with weak annotations outperform the fully supervised SE(VOC) and do not reach the performance of HED(VOC). As has been shown in Section 4 the HED detector is less robust to noise than SE.

Det.+HED $(\cdot)$ Combining an object detector with HED(VOC) (see Det.+HED (VOC) in Figure 7) is not beneficial to the performance as the HED detector already has notion of objects and their location due to pixel-to-pixel end-to-end learning of the network.

For HED models trained with the weakly supervised variants, employing an object detector at test time brings only a slight improvement of the performance in the high precision area. The reason for this is that we already use information from the bounding box detector to generate the annotation and the convnet method is able to learn it during training.

Det.+HED (MCG $\cap$ BBs) outperforms Det.+HED (BSDS) (see Table 3). Note that the HED trained with the proposed annotations, generated without using boundary ground truth, performs on par with the HED model trained on generic boundaries (Det.+HED (cons. S&G$\cap$BBs) and Det.+HED (BSDS)in Figure 7).

The qualitative results are presented in Figure 8 and support the quantitative evaluation.

Conclusion Similar to other computer vision tasks deep convnet methods show superior performance. Due to the

	Family	Method	mF	mAP
Other	GT	Hariharan et al. [16]	28	21
SE	GT	SB(SBD) orig. [31]	39	32
		SB(SBD)	43	37
		Det.+SE (SBD)	51	45
	Other GT	Det.+SE (BSDS)	51	44
		Det.+MCG (BSDS)	50	42
	Weakly super-vised	SB(SeSe ∩ BBs)	40	34
		SB (MCG ∩ BBs)	42	35
		Det.+SE (SeSe ∩ BBs)	48	42
		Det.+SE (MCG ∩ BBs)	**51**	**45**
HED	GT	HED (SBD)	44	41
		Det.+HED (SBD)	49	45
	Other GT	HED(BSDS)	38	32
		Det.+HED (BSDS)	49	44
	Weakly super-vised	HED(cons. MCG ∩ BBs)	41	37
		HED (cons. S&G ∩ BBs)	44	39
		Det.+HED (cons. MCG ∩ BBs)	48	44
		Det.+HED (cons. S&G ∩ BBs)	**52**	**47**

Table 5: SBD results. Results are mean F(ODS)/AP across all 20 categories. (·) denotes the data used for training. See also Figure 9. Bold indicates our best weakly supervised results.

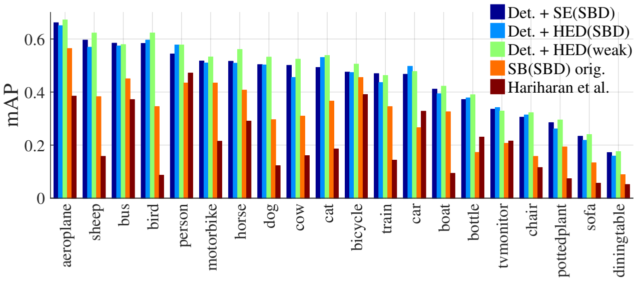


Figure 9: SBD results per class. (·) denotes the data used for training. Det.+HED (weak) refers to the model Det.+ HED (cons. S&G ∩ BBs).

pixel-to-pixel training and global view of the image the convnet models have a notion of object and its location which allows to omit the use of the detector at test time. With our weakly supervised boundary annotations we can gain fair performance without using any instance-wise object boundary or generic boundary annotations and leave out object detection at test time by feeding object bounding box information during training.

8. COCO boundary detection results

Additionally we show the generalization of the proposed weakly supervised variants for object boundary detection on the COCO dataset. We use the same evaluation protocol as for VOC. For weakly supervised cases the results are shown with the models trained on VOC, without re-training on COCO.

The results are summarized in Table 4. On the COCO benchmark for both SE and HED the models trained on the proposed weak annotations perform as well as the fully supervised SE models. Similar to the VOC benchmark the HED model trained on ground truth shows superior performance.

9. SBD boundary detection results

In this section we analyse the performance of the proposed weakly supervised boundary variants trained with SE and HED on the SBD dataset [16]. In contrast to the VOC benchmark we move from object boundaries to class specific object boundaries. We are interested in external boundaries of all annotated objects of the specific semantic class and all internal boundaries are ignored during evaluation following the benchmark [16]. The results are presented in Figure 9 and in Table 5.

Fully supervised Applying SE model plus object detection at test time outperforms the class specific situational boundary detector (for both [31] and our re-implementation) as well as the Inverse Detectors [16]. The model trained with SE on ground truth performs as well as the HED detector. Both of the models are good at detecting external object boundaries; however SE, being a more local, triggers more on internal boundaries than HED. In the VOC evaluation detecting internal object boundaries is penalized, while in SBD these are ignored. This explains the small gap in the performance between SE and HED on this benchmark.

Weakly supervised The models trained with the proposed weakly-supervised boundary variants perform on par with the fully supervised detectors, while only using bounding boxes or generic boundary annotations. We show in Table 5 the top result with the Det. + HED(cons. S&G∩BBs) model, achieving the state-of-the-art performance on the SBD benchmark. As Figure 9 shows our weakly supervised approach considerably outperforms [31, 16] on all 20 classes.

Conclusion

The presented experiments show that when using the bounding box annotations for training an object detector, one can also train a high quality object boundary detector without additional annotation effort.

Using boxes alone, our proposed weak-supervision techniques improve over previously reported fully supervised results for object-specific boundaries. When using generic boundary or ground truth annotations, we also achieve the top performance on the object boundary detection task, outperforming previously reported results by a large margin.

To facilitate future research all the resources of this project - source code, trained models and results - will be made publicly available.

Acknowledgements We thank J. Hosang for help with Fast R-CNN training; S. Oh and B. Pepik for valuable feedback.

References

[1] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *PAMI*, 2011. 1, 2, 4

[2] D. Banica and C. Sminchisescu. Second-order constrained parametric proposals and sequential search-based structured prediction for semantic segmentation in RGB-D images. In *CVPR*, 2015. 1

[3] G. Bertasius, J. Shi, and L. Torresani. Deepedge: A multi-scale bifurcated deep network for top-down contour detection. In *CVPR*, 2015. 2

[4] G. Bertasius, J. Shi, and L. Torresani. High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision. In *ICCV*, 2015. 2

[5] G. Bertasius, J. Shi, and L. Torresani. Semantic segmentation with boundary neural fields. In *CVPR*, 2016. 2

[6] J. Canny. A computational approach to edge detection. *PAMI*, 1986. 2, 3

[7] C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, W. Xu, D. Ramanan, and T. Huang. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *ICCV*, 2015. 2

[8] M.M. Cheng, V. Prisacariu, S. Zheng, P. Torr, and C. Rother. Densecut: Densely connected crfs for real-time grabcut. *Computer Graphics Forum*, 2015. 4

[9] P. Dollár and C. L. Zitnick. Fast edge detection using structured forests. *PAMI*, 2015. 1, 2, 3

[10] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 2015. 1, 2

[11] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004. 1, 2, 3, 4, 5

[12] F. Galasso, N.S. Nagaraja, T. Jimenez, T. Brox, and B. Schiele. A unified video segmentation benchmark: Annotation, metrics and analysis. In *ICCV*, 2013. 1

[13] Y. Ganin and V. Lempitsky. N4-fields: Neural network nearest neighbor fields for image transforms. In *ACCV*, 2014. 2

[14] R. Girshick. Fast R-CNN. In *ICCV*, 2015. 1, 4, 5, 6

[15] S. Hallman and C. Fowlkes. Oriented edge forests for boundary detection. In *CVPR*, 2015. 2

[16] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In *ICCV*, 2011. 1, 2, 3, 8

[17] J. Hosang, R. Benenson, P. Dollár, and B. Schiele. What makes for effective detection proposals? *PAMI*, 2015. 1

[18] X. Hou, A. Yuille, and C. Koch. Boundary detection benchmarking: Beyond f-measures. In *CVPR*, 2013. 3

[19] P. Isola, D. Zoran, D. Krishnan, and E. H. Adelson. Crisp boundary detection using pointwise mutual information. In *ECCV*, 2014. 2

[20] Y. Li, M. Paluri, J. M. Rehg, and P. Dollár. Unsupervised learning of edges. In *CVPR*, 2016. 2

[21] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2, 3

[22] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001. 2

[23] M. Oquab, L. Bottou, Laptev I, and Sivic J. Is object localization for free? – weakly-supervised learning with convolutional neural networks. In *CVPR*, 2015. 2

[24] P. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional network. In *CVPR*, 2015. 2

[25] J. Pont-Tuset, P. Arbeláez, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping for image segmentation and object proposal generation. In *arXiv:1503.00848*, March 2015. 1, 2, 5

[26] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012. 2

[27] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 1, 4

[28] C. Rother, V. Kolmogorov, and A. Blake. Grabcut - interactive foreground extraction using iterated graph cuts. *SIGGRAPH*, 2004. 4

[29] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR Workshop*, 2014. 4

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 3, 6

[31] J.R.R. Uijlings and V. Ferrari. Situational object boundary detection. In *CVPR*, 2015. 2, 6, 8

[32] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders. Selective search for object recognition. *IJCV*, 2013. 1, 5

[33] A. Vezhnevets, V. Ferrari, and J.M. Buhmann. Weakly supervised semantic segmentation with a multi-image model. In *ICCV*, 2011. 2

[34] C. Wang, W. Ren, K. Huang, and T. Tan. Weakly supervised object localization with latent category learning. In *ECCV*, 2014. 2

[35] S. Xie and Z. Tu. Holistically-nested edge detection. In *ICCV*, 2015. 1, 2, 3, 4, 7

[36] J. Xu, A. Schwing, and R. Urtasun. Learning to segment under various weak supervisions. In *CVPR*, 2015. 2

[37] Y. Zhu, R. Urtasun, R. Salakhutdinov, and S. Fidler. segdeepm: Exploiting segmentation and context in deep neural networks for object detection. In *CVPR*, 2015. 1