Simple Does It: Weakly Supervised Instance and Semantic Segmentation

Anna Khoreva[1] Rodrigo Benenson[1] Jan Hosang[1] Matthias Hein[2] Bernt Schiele[1]

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

1. Content

This supplementary material provides additional quantitative and qualitative results:

- Section 2 analyses the contribution of the post-processing stages during recursive training (Figure S1).
- Section 3 discusses training differences of our approach in contrast to the related work.
- We report a comparison of different GrabCut-like methods on Pascal VOC12 boxes in Section 4.
- Section 5 (Figure S2) shows visualization of the different variants of the proposed segmentation inputs obtained from bounding box annotations for weakly supervised semantic segmentation.
- Detailed performance of each class for semantic labelling is reported in Section 6 (Table S2).
- Section 7 provides additional qualitative results for weakly supervised semantic segmentation on Pascal VOC12 (Figure S3).
- Qualitative results for instance segmentation are shown in Section 8 (Figure S4 and Figure S5).

2. Recursive training with boxes

In Section 3 of the main paper we recursively train a convnet directly on the full extend of bounding box annotations as foreground labels, disregarding post-processing stages. We name this recursive training approach `Naive`. Using this supervision and directly applying recursive training leads to significant degradation of the segmentation output quality, see Figure S1.

To improve the labels between the training rounds three post-processing stages are proposed. Here we discuss them in more detail:

1. **Box enforcing**: Any pixel outside the box annotations is reset to background label (cue C1, see Section 3 in the main paper).

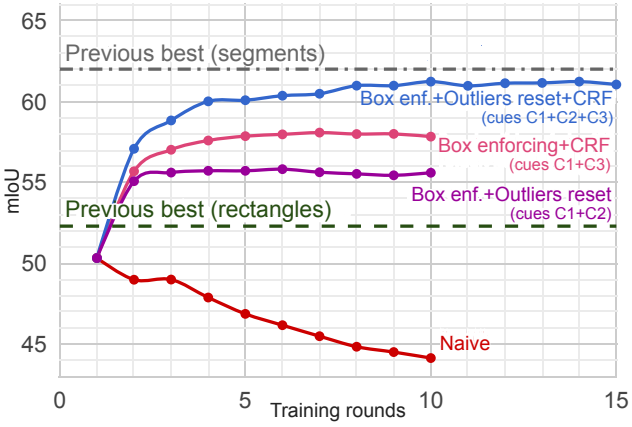

Figure S1: Recursive training from rectangles only as input. Validation set results. All methods use only rectangles as initial input, except "previous best (segments)".

2. **Outliers reset**: If the area of a segment is too small compared to its corresponding bounding box (e.g. IoU$<$ 50%), the box area is reset to its initial label (fed in the first round). This enforces a minimal area (cue C2).
3. **CRF**: As it is common practice among semantic labelling methods, we filter the output of the network to better respect the image boundaries. (We use DenseCRF [5] with the DeepLabv1 parameters [2]). In our weakly supervised scenario, boundary-aware filtering is particularly useful to improve objects delineation (cue C3).

Results Figure S1 presents results of the recursive training using boxes as input and shows the contribution of the post-processing stages. We see that the naive recursive training is ineffectual. However as soon as some constraints (box enforcing and outliers reset, cues C1+C2) are enforced, the quality improves dramatically after the first round of recursive training. These results already improve over previous work considering rectangles only input [4, 6] (both using a similar convnet to ours) and achieve 3 points improvement over [6] (from 52.5 to 55.6 mIoU, see Figure

S1 "Box enf.+Outliers reset").

Even more, when also adding CRF filtering (+ cue C3) over the training set, we see a steady grow after each round, stabilizing around 61% mIoU. This number is surprisingly close to the best results obtained using more sophisticated techniques [4], which achieve around 62% mIoU (see Figure S1 and Table S2).

Our results indicate that recursive training of a convnet is robust to input noise as soon as appropriate care is taken to de-noise the output between rounds, enabled by given bounding boxes and object priors.

3. Training details in comparison with BoxSup and WSSL

In this work we focus on box level annotations for semantic labelling of objects. The closest related work are thus [4, 6]. Since all implementations use slightly different networks and training procedures, care should be taken during comparison. Both [4] and [6] propose new ways to train convnets under weak supervision. Both of the approaches build upon the DeepLab network [2], however, there are a few differences in the network architecture.

WSSL [6] employs 2 different variants of the DeepLab architecture with small and large receptive field of view (FOV) size. For each experiment WSSL evaluates with both architectures and reports the best result obtained (using boxes or segments as input). BoxSup [4] uses their own implementation of the DeepLab with the small FOV. In our approach all the experiments employ the DeepLab architecture with the large FOV.

There are also differences in the training procedure. For SGD WSSL uses a mini-batch of 20-30 images and fine-tunes the network for about 12 hours (number of epochs is not specified) with the standard learning parameters (following [2]). In the SGD training BoxSup uses a mini-batch size of 20 and the learning rate is divided by 10 after every 15 epochs. The training is terminated after 45 epochs. We use a mini-batch of 30 images for SGD and the learning rate is divided by 10 after every 2k iterations, ~6 epochs. Our network is trained for 6k iterations, ~18 epochs.

Similarly to our approach, the BoxSup method [4] uses MCG object proposals during training. However, there are important differences. They modify the training procedure so as to denoise intermediate outputs by randomly selecting high overlap proposals. In comparison, our approach keeps the training procedure unmodified and simply generates input labels. Our approach also uses ignore regions, while BoxSup does not explore this dimension.

WSSL [6] proposes an expectation-maximisation algorithm with a bias to enable the network to estimate the foreground regions. In contrast, in our work we show that one can reach better results without modifying the training procedure (compared to the fully supervised case) by instead carefully generating input labels for training from the bounding box annotations (Section 3.2 in the main paper).

4. GrabCut variants

As discussed in Section 3.2 in the main paper we propose to employ box-guided instance segmentation to increase quality of the input data. Our goal is to have weak annotations with maximal quality and minimal loss in recall. In Section 3.1 in the main paper we explored how far could we get with just using boxes as foreground labels. However, to obtain results of higher quality several rounds of recursive training are needed. Starting from less noisier object segments we would like to reach better performance with just one training round.

For this purpose we explore different GrabCut-like [7] techniques, the corresponding quantitative results are in Table S1. For evaluation we use the mean IoU measure. Previous work evaluated using the 50 images from the GrabCut dataset [7], or 1k images with one salient object [3]. The evaluation of Table S1 compares multiple methods over 3.4k object windows, where the objects are not salient, have diverse sizes and occlusions level. This is a more challenging scenario than usually considered for GrabCut-like methods.

	Method	mIoU
GrabCut variants	DenseCut [3]	52.5
	Bbox-Seg+CRF [6]	71.1
	GrabCut [7]	72.9
	KGrabCut [8]	73.5
	GrabCut+	75.2

Table S1: GrabCut variants, evaluated on Pascal VOC12 validation set. See Section 4 for details.

`GrabCut` [7] is the established technique to estimate an object segment from its bounding box. To further improve its quality we propose to use better pairwise terms. We name this variant `GrabCut+`. Instead of the typical RGB colour difference the pairwise terms in `GrabCut+` are replaced by probability of boundary as generated by HED [9]. The HED boundary detector is trained on the generic boundaries of BSDS500 [1]. Moving from `GrabCut` to `GrabCut+` brings a ~ 2 points improvement, see Table S1.

We also experimented with other variants such as `DenseCut` [3] and `KGrabCut` [8] but did not obtain significant gains.

[6] proposed to perform foreground/background segmentation by using DenseCRF and the 20% of the centre area of the bounding box as foreground prior. This ap-

proach is denoted `Bbox-Seg+CRF` in Table S1 and underperforms compared to `GrabCut` and `GrabCut+`.

5. Examples of input segmentations

Figure S2 presents examples of the considered weak annotations. This figure extends Figure 3 of the main paper.

6. Detailed test set results for semantic labelling

In Table S2, we present per class results on the Pascal VOC12 test set for the methods reported in the main paper in Table 2.

On average with our weakly supervised results we achieve $\sim 95\%$ quality of full supervision across all classes when training with VOC12 only or VOC12+COCO.

7. Qualitative results for semantic labelling

Figure S3 presents qualitative results for semantic labelling on Pascal VOC12. The presented semantic labelling examples show that high quality segmentation can be achieved using only detection bounding box annotations. This figure extends Figure 5 of the main paper.

8. Qualitative results for instance segmentations

Figure S4 illustrates additional qualitative results for instance segmentations given by the weakly supervised DeepMask and $\text{DeepLab}_{\text{BOX}}$ models. This figure complements Figure 6 from the main paper.

Figure S5 shows examples of instance segmentation given by different methods. Our proposed weakly supervised DeepMask model achieves competitive performance with fully supervised results and provides higher quality output in comparison with box-guided segmentation techniques. The $\text{DeepLab}_{\text{BOX}}$ model also provides similar results, see Table 4 in the main paper.

Training data	Super-vision	Method	mean	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor bike	per son	plant	sheep	sofa	train	tv
VOC12	weak	Box	62.2	62.6	24.5	63.7	56.7	68.1	84.3	75.0	72.3	27.2	63.5	61.7	68.2	56.0	70.9	72.8	49.0	66.7	45.2	71.8	58.3
		Box^{i}	63.5	67.7	25.5	67.3	58.0	62.8	83.1	75.1	78.0	25.5	64.7	60.8	74.0	62.9	74.6	73.3	50.0	68.5	43.5	71.6	56.7
		M $\cap$ G+	**67.5**	78.1	31.1	72.4	61.0	67.2	84.2	78.2	81.7	27.6	68.5	62.1	76.9	70.8	78.0	76.3	51.7	78.3	48.3	74.2	58.6
	semi	M $\cap$ G+	66.9	75.8	32.3	75.9	60.1	65.7	82.9	75.0	79.5	29.5	68.5	60.6	76.2	68.6	76.9	75.2	53.2	76.6	49.5	73.8	58.6
	full	WSSL [6]	70.3	83.5	36.6	82.5	62.3	66.5	85.4	78.5	83.7	30.4	72.9	60.4	78.5	75.5	82.1	79.7	58.2	82.0	48.8	73.7	63.3
		DeepLab$_{ours}$ [2]	70.5	85.3	38.3	79.4	61.4	68.9	86.4	82.1	83.6	30.3	74.5	53.8	78.0	77.0	83.7	81.8	55.6	79.8	45.9	79.3	63.4
VOC12 + COCO	weak	Box^{i}	66.7	69.0	27.5	77.1	61.9	65.3	84.2	75.5	83.2	25.7	73.6	63.6	78.2	69.3	75.3	75.2	51.0	73.5	46.2	74.4	60.4
		M $\cap$ G+	**69.9**	82.5	33.4	82.5	59.5	65.8	85.3	75.6	86.4	29.3	77.1	60.8	80.7	79.0	80.5	77.6	55.9	78.4	48.6	75.2	61.5
	semi	BoxSup [4]	71.0	86.4	35.5	79.7	65.2	65.2	84.3	78.5	83.7	30.5	76.2	62.6	79.3	76.1	82.1	81.3	57.0	78.2	55.0	72.5	68.1
		M $\cap$ G+	**72.8**	87.6	37.7	86.7	65.5	67.3	86.8	81.1	88.3	30.7	77.3	61.6	82.7	79.4	84.1	82.0	60.3	84.0	49.4	77.8	64.7
	full	WSSL [6]	72.7	89.1	38.3	88.1	63.3	69.7	87.1	83.1	85.0	29.3	76.5	56.5	79.8	77.9	85.8	82.4	57.4	84.3	54.9	80.5	64.1
		DeepLab$_{ours}$ [2]	73.2	88.8	37.3	83.8	66.5	70.1	89.0	81.4	87.3	30.2	78.8	61.6	82.4	82.3	84.4	82.2	59.1	85.0	50.8	79.7	63.8

Table S2: Per class semantic labelling results for methods trained using Pascal VOC12 and COCO. Test set results. Bold indicates the best performance with the same supervision and training data. M $\cap$ G+ denotes the weakly or semi supervised model trained with MCG $\cap$ Grabcut+.

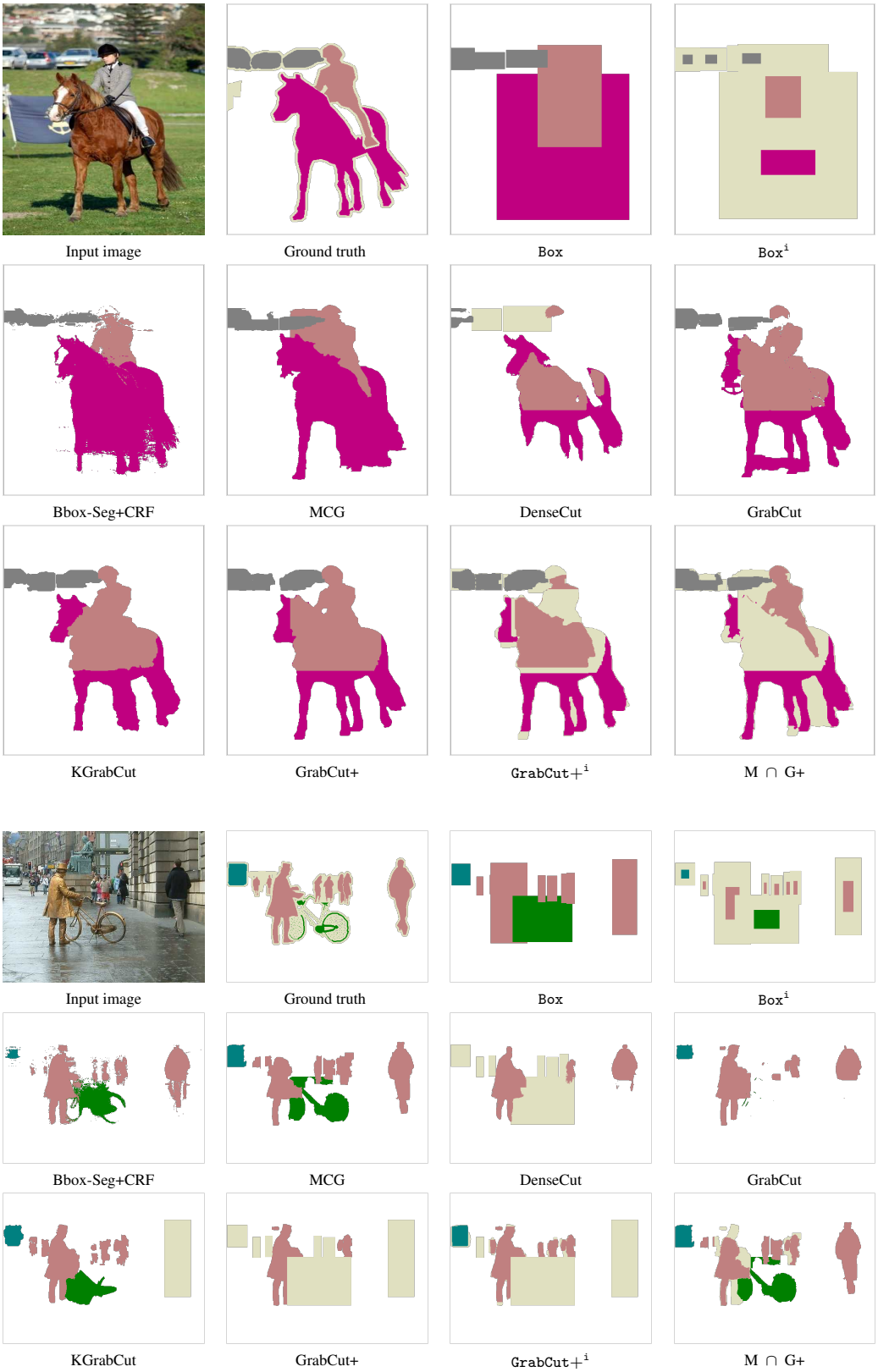

Figure S2: Different segmentations obtained starting from a bounding box. White is background and ignore regions are beige. M ∩ G+ denotes MCG ∩ Grabcut+.

Figure S2: Different segmentations obtained starting from a bounding box. White is background and ignore regions are beige. M ∩ G+ denotes MCG ∩ Grabcut+.

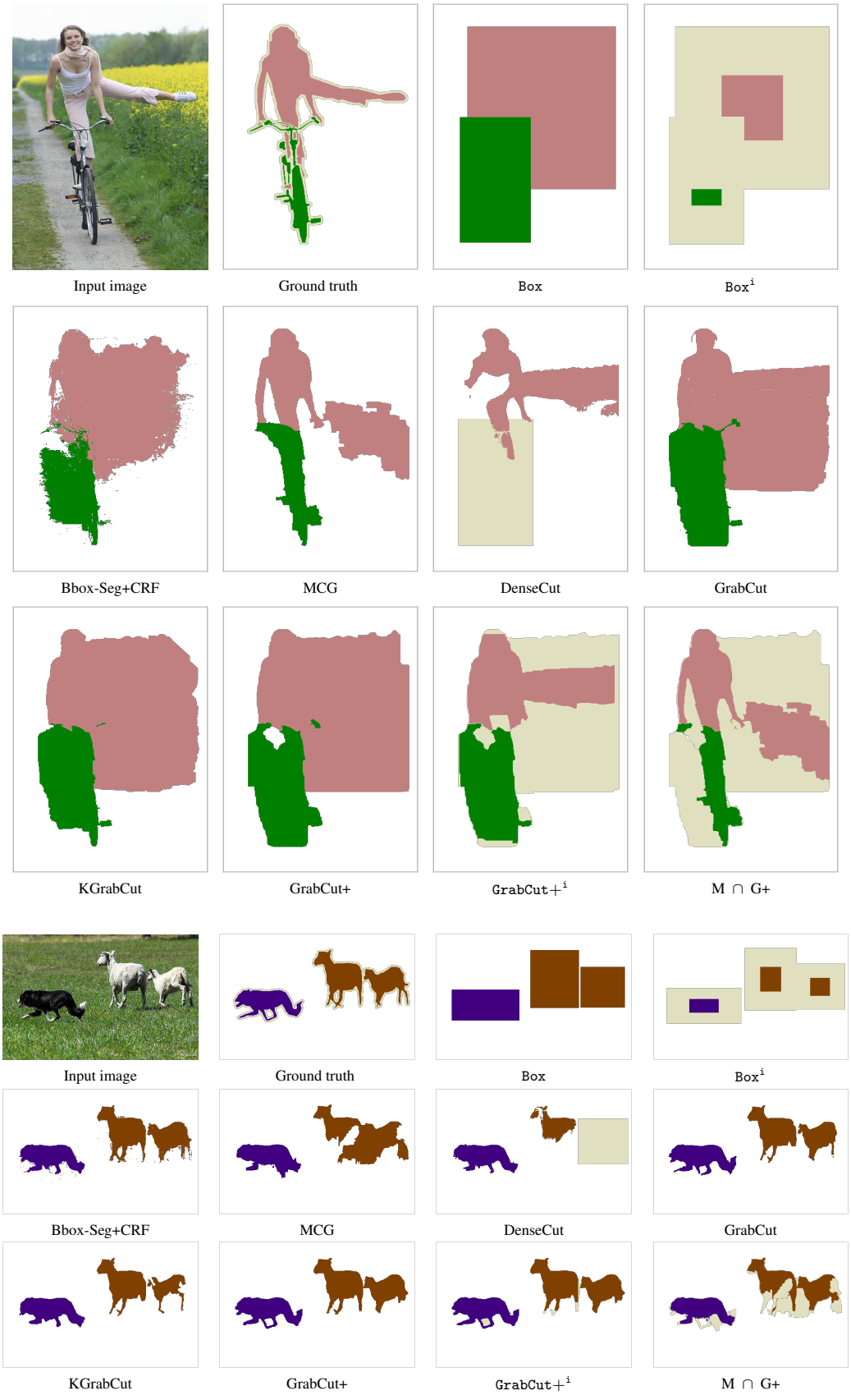

Figure S2: Different segmentations obtained starting from a bounding box. White is background and ignore regions are beige. M ∩ G+ denotes MCG ∩ Grabcut+.

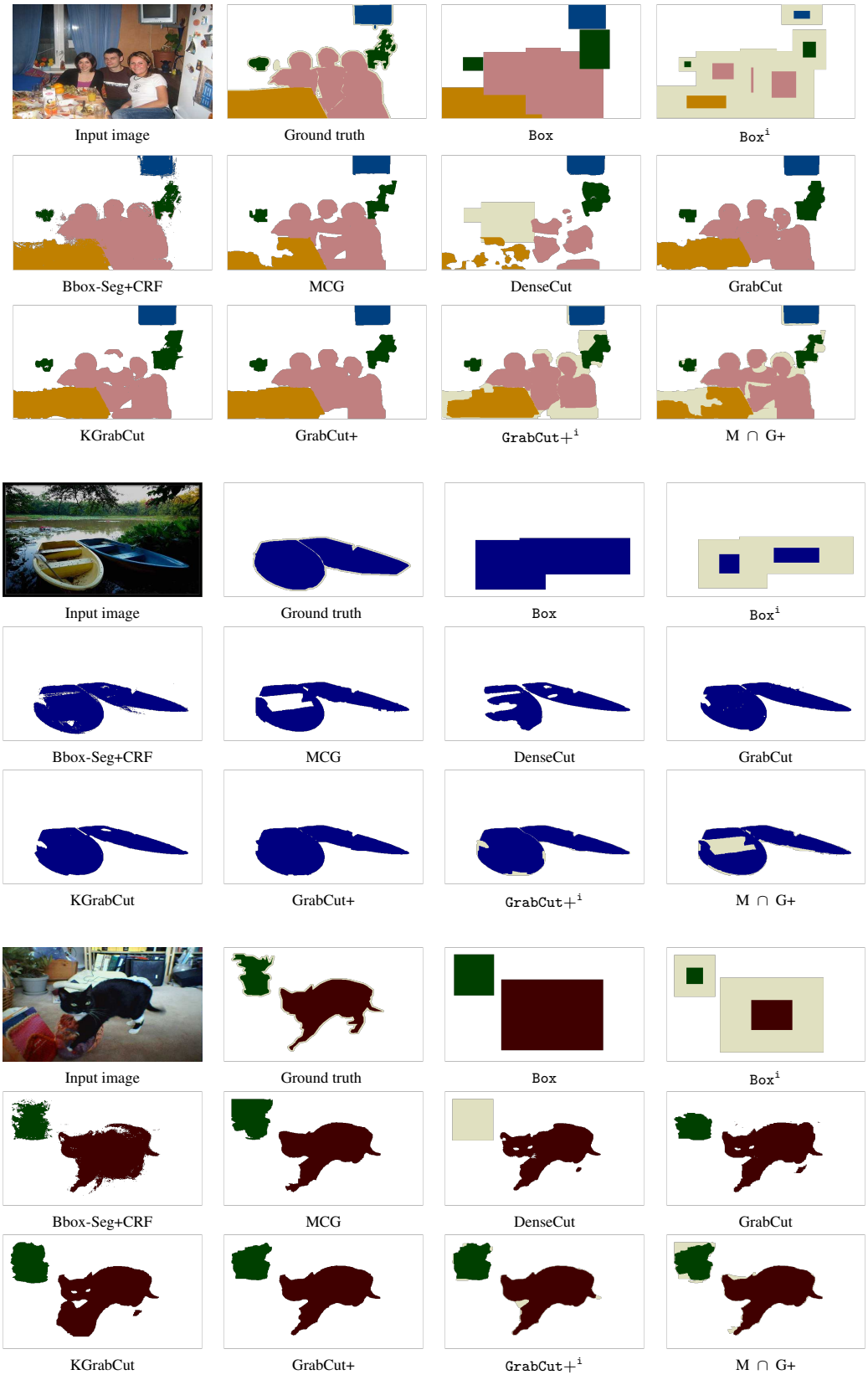

Figure S2: Different segmentations obtained starting from a bounding box. White is background and ignore regions are beige. M ∩ G+ denotes MCG ∩ Grabcut+.

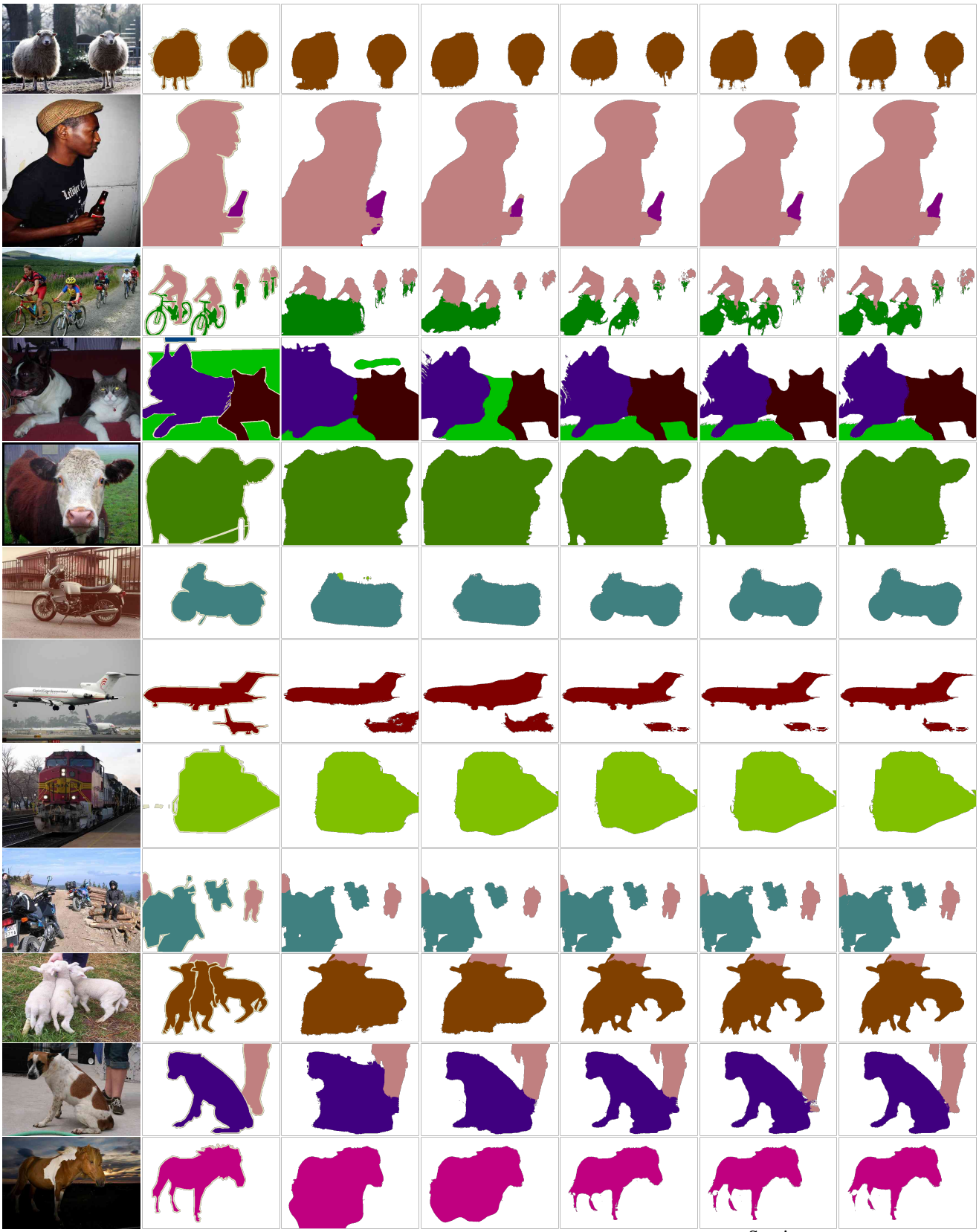

Figure S3: Qualitative results on VOC12. M ∩ G+ denotes the weakly supervised model trained on MCG ∩ Grabcut+.

DeepMask

$\text{DeepLab}_{\text{BOX}}$

Figure S4: Example results from the DeepMask and $\text{DeepLab}_{\text{BOX}}$ models trained with Pascal VOC12 and COCO using box supervision. White boxes illustrate Fast-RCNN detection proposals used to output the segments which have the best overlap with the ground truth segmentation mask.

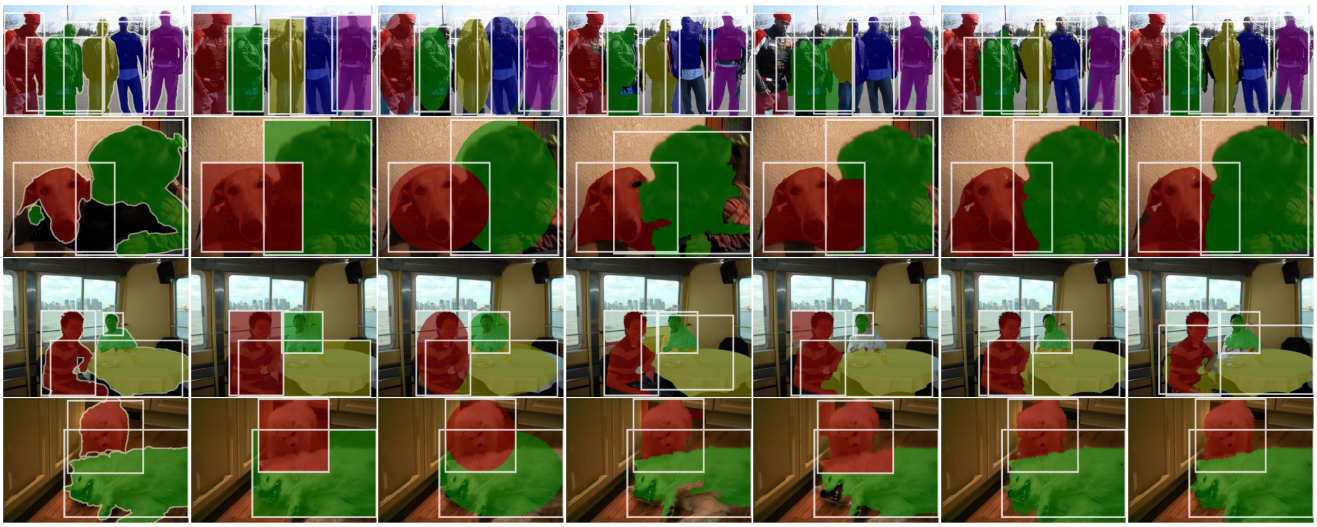

Figure S5: Qualitative results of instance segmentation on VOC12. Example result from the DeepMask model are trained with Pascal VOC12 and COCO supervision. White boxes illustrate Fast-RCNN detection proposals used to output the segments which have the best overlap with the ground truth segmentation mask.

References

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

[2] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 1, 2, 4

[3] M. Cheng, V. Prisacariu, S. Zheng, P. Torr, and C. Rother. Densecut: Densely connected crfs for realtime grabcut. *Computer Graphics Forum*, 2015. 2

[4] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, 2015. 1, 2, 4

[5] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*. 2011. 1

[6] G. Papandreou, L. Chen, K. Murphy, , and A. L. Yuille. Weakly- and semi-supervised learning of a dcnn for semantic image segmentation. In *ICCV*, 2015. 1, 2, 4

[7] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM Trans. Graphics*, 2004. 2

[8] M. Tang, I. Ben Ayed, D. Marin, and Y. Boykov. Secrets of grabcut and kernel k-means. In *ICCV*, 2015. 2

[9] S. Xie and Z. Tu. Holistically-nested edge detection. In *ICCV*, 2015. 2