Appendix

In this section, we provide: (1) additional quantitative results on COCO; (2) per-class detection (AP) and correct localization (CorLoc) results on VOC; (3) additional qualitative results; (4) proposal statistics; (5) ablation study on the amount of proposals; (6) implementation details and video demo of weakly supervised video object detection. Specifically, we show that our approach produces state-of-the-art results on COCO (see Tab. 8), outperforms all competing models on VOC 2007 and 2012 (see Tab. 9 and Tab. 10). We also provide correct localization results in Tab. 11 and Tab. 12 for completeness and illustrate the necessity of the sequential batch back-propagation (introduced in Sec. 4.3 of the main paper) in Tab. 13 and Tab. 14. Comprehensive visualizations are also provided (Fig. 13 to Fig. 16).

A. Additional quantitative results on COCO

In Tab. 8, we report quantitative results at different thresholds and scales on COCO for different models. The reported metrics include: Average Prevision (AP) over multiple IoU thresholds (.50 : .05 : .95), at IoU threshold 50% and 75% (AP^{50}, AP^{75}), and for small, medium and large objects (AP^{s}, AP^{m}, AP^{l}); and Average Recall (AR) over multiple IoU values (.50 : .05 : .95), given 1, 10 and 100 detections per image (AR^{1}, AR^{10}, AR^{100}); and for small, medium and large objects (AR^{s}, AR^{m}, AR^{l}). The results in Tab. 8 show that object size is a significant factor that influences the detection accuracy. The detector tends to perform better on large objects rather than smaller ones.

B. Additional results on VOC

B.1. Per-class detection results

In Tab. 9 and Tab. 10, we report the per-class detection APs on the test sets of both VOC 2007 and 2012. Compared to other WSOD methods we observe: (1) Our method outperforms all others on most categories (10 classes on VOC 2007, 14 classes on VOC 2012). (2) The classes that are hard for our approach (*e.g.*, boat, plant, and chair) are also challenging for other methods. This suggests that these categories are essentially hard examples for WSOD methods, for which a certain amount of strong supervision might still be needed.

Compared to supervised models (Fast R-CNN, Faster R-CNN) we note: (1) Our weakly supervised model performs competitively for classes such as: airplane, bicycle, bus, car, cow, motorbike, sheep, tv-monitor, where the performance gap is usually less than 10% AP. Our model sometimes even outperforms supervised models on categories that are considered relatively easy with small intra-class difference (bicycle and motorbike in VOC 2007, motorbike and tv-monitor in VOC 2012). (2) For classes like boat, chair, dinning table, person, all WSOD methods are significantly worse than supervised methods. This is likely due to a large intra-class variation. WSOD methods fail to capture the consistent patterns of these classes.

B.2. Per-class correct localization results

In Tab. 11 and Tab. 12, we report the per-class correct localization (CorLoc) results on the trainval sets of both VOC 2007 and VOC 2012. Consistent with prior work [5, 45, 50, 60, 62, 2] this metric is computed on the training set. Thus it does not reflect the true performance of the detection models and has not been widely adopted by supervised methods [15, 35, 17]. For WSOD approaches, it serves as an indicator of the 'over-fitting' behavior. Compared with previous state-of-the-art, our method achieves the third best result on VOC 2007, winning on 2 categories. We also achieve the second best performance on VOC 2012 and win on 19 categories. We find that: (1) Our model performs well for classes like: airplane, bicycle, bottle, bus, motorbike, sheep, tv-monitor. This observation aligns very well with the detection results. (2) The best performing methods differ across classes, which suggest that methods could potentially be ensembled for further improvements.

C. Additional qualitative results

C.1. Results on static-image datasets

We show additional results that highlight cases of *'Instance Ambiguity'* and *'Part Domination'* in Fig. 13 and Fig. 14, respectively. Following the main paper, we compare our final model to a baseline without the modules proposed in Sec. 4.1 and Sec. 4.2 of the main paper to demonstrate the effectiveness of these two modules visually. We show a set of two pictures side by side, the baseline on the left and ours on the right. From the results, we observe: (1) we have addressed the *'Missing Instances'* issue and previously ignored objects are detected with great recall (*e.g.*, monitor, sheep, car, and person in Fig. 13); (2) we have addressed the *'Grouped Instances'* issue as our model predicts tight and precise boxes for multiple instances rather than one big one (*e.g.*, bus, motor, boat, car in Fig. 13); (3) we have also alleviated the *'Part Domination'* issue for objects like dog, cat, sheep, person, horse, and sofa (see Fig. 14).

We also provide additional visualization of our results on COCO in Fig. 15. We obtain these results by running the VGG16 based model on the COCO 2014 validation set. Our model is able to detect different instances of the same category (*e.g.*, car, elephant, pizza, cow, umbrella) and various objects of different classes in relatively complicated scenes, and the obtained boxes can cover the whole objects pretty well rather than simply focusing on discriminative parts.

[†] http://host.robots.ox.ac.uk:8080/anonymous/DCJ5GA.html

Train	Test	Model	AP	AP^{50}	AP^{75}	AP^{s}	AP^{m}	AP^{l}	AR^{1}	AR^{10}	AR^{100}	AR^{s}	AR^{m}	AR^{l}
2014 Train	2014 Val	VGG16	11.4	24.3	9.4	3.6	12.2	17.6	13.5	22.6	23.9	8.5	25.4	38.3
2014 Train	2014 Val	R50-C4	12.6	26.1	10.8	3.7	13.3	19.9	14.8	23.7	24.7	8.4	25.1	41.8
2014 Train	2014 Val	R101-C4	13.0	26.3	11.4	3.5	13.7	20.4	15.4	23.4	24.6	8.5	24.6	40.9
2017 Train	minival	VGG16	12.4	25.8	10.5	3.9	13.8	19.9	14.3	23.3	24.6	9.7	26.6	39.6
2014 Train	Test-Dev	VGG16	12.1	24.8	10.2	4.1	13.0	18.3	13.5	25.5	29.0	9.6	30.0	46.7

Table 8: Single model detection results on COCO.

Methods	Proposal	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Motor	Person	Plant	Sheep	Sofa	Train	TV	AP
Fast R-CNN	SS	73.4	77.0	63.4	45.4	44.6	75.1	78.1	79.8	40.5	73.7	62.2	79.4	78.1	73.1	64.2	35.6	66.8	67.2	70.4	71.1	66.0
Faster R-CNN	RPN	70.0	80.6	70.1	57.3	49.9	78.2	80.4	82.0	52.2	75.3	67.2	80.3	79.8	75.0	76.3	39.1	68.3	67.3	81.1	67.6	**69.9**
Cinbis [7]	SS	35.8	40.6	8.1	7.6	3.1	35.9	41.8	16.8	1.4	23.0	4.9	14.1	31.9	41.9	19.3	11.1	27.6	12.1	31.0	40.6	22.4
Bilen [4]	SS	46.2	46.9	24.1	16.4	12.2	42.2	47.1	35.2	7.8	28.3	12.7	21.5	30.1	42.4	7.8	20.0	26.8	20.8	35.8	29.6	27.7
Wang [52]	SS	48.8	41.0	23.6	12.1	11.1	42.7	40.9	35.5	11.1	36.6	18.4	35.3	34.8	51.3	17.2	17.4	26.8	32.8	35.1	45.6	30.9
Li [27]	EB	54.5	47.4	41.3	20.8	17.7	51.9	63.5	46.1	21.8	57.1	22.1	34.4	50.5	61.8	16.2	**29.9**	40.7	15.9	55.3	40.2	39.5
WSDDN [5]	EB	39.4	50.1	31.5	16.3	12.6	64.5	42.8	42.6	10.1	35.7	24.9	38.2	34.4	55.6	9.4	14.7	30.2	40.7	54.7	46.9	34.8
Teh [47]	EB	48.8	45.9	37.4	26.9	9.2	50.7	43.4	43.6	10.6	35.9	27.0	38.6	48.5	43.8	24.7	12.1	29.0	23.2	48.8	41.9	34.5
ContextLocNet [22]	SS	57.1	52.0	31.5	7.6	11.5	55.0	53.1	34.1	1.7	33.1	49.2	42.0	47.3	56.6	15.3	12.8	24.8	48.9	44.4	47.8	36.3
OICR [45]	SS	58.0	62.4	31.1	19.4	13.0	65.1	62.2	28.4	24.8	44.7	30.6	25.3	37.8	65.5	15.7	24.1	41.7	46.9	64.3	62.6	41.2
Jie [21]	?	52.2	47.1	35.0	26.7	15.4	61.3	66.0	54.3	3.0	53.6	24.7	43.6	48.4	65.8	6.6	18.8	51.9	43.6	53.6	62.4	41.7
Diba [9]	EB	49.5	60.6	38.6	29.2	16.2	70.8	56.9	42.5	10.9	44.1	29.9	42.2	47.9	64.1	13.8	23.5	45.9	54.1	60.8	54.5	42.8
PCL [44]	SS	54.4	69.0	39.3	19.2	15.7	62.9	64.4	30.0	25.1	52.5	44.4	19.6	39.3	67.7	17.8	22.9	46.6	57.5	58.6	63.0	43.5
Wei [56]	SS	59.3	57.5	43.7	27.3	13.5	63.9	61.7	59.9	24.1	46.9	36.7	45.6	39.9	62.6	10.3	23.6	41.7	52.4	58.7	56.6	44.3
Tang [46]	SS	57.9	70.5	37.8	5.7	21.0	66.1	69.2	59.4	3.4	57.1	**57.3**	35.2	64.2	68.6	32.8	28.6	50.8	49.5	41.1	30.0	45.3
Shen [37]	SS	52.0	64.5	45.5	26.7	27.9	60.5	47.8	59.7	13.0	50.4	46.4	56.3	49.6	60.7	25.4	28.2	50.0	51.4	66.5	29.7	45.6
Wan [51]	SS	55.6	66.9	34.2	29.1	16.4	68.8	68.1	43.0	25.0	65.6	45.3	53.2	49.6	68.6	2.0	25.4	52.5	56.8	62.1	57.1	47.3
SDCN [28]	SS	59.4	71.5	38.9	32.2	21.5	67.7	64.5	**68.9**	20.4	49.2	47.6	60.9	55.9	67.4	31.2	22.9	45.0	53.2	60.9	64.4	50.2
C-MIL [50]	SS	62.5	58.4	49.5	32.1	19.8	70.5	66.1	63.4	20.0	60.5	52.9	53.5	57.4	68.9	8.4	24.6	51.8	58.7	66.7	63.6	50.5
Yang [59]	SS	57.6	70.8	50.7	28.3	27.2	72.5	69.1	65.0	26.9	64.5	47.4	47.7	53.5	66.9	13.7	29.3	56.0	54.9	63.4	65.2	51.5
C-MIDN [12]	SS	53.3	71.5	49.8	26.1	20.3	70.3	69.9	68.3	28.7	65.3	45.1	**64.6**	58.0	71.2	20.0	27.5	54.9	54.9	**69.4**	63.5	52.6
Arun [2]	SS	66.7	69.5	52.8	31.4	24.7	**74.5**	74.1	67.3	14.6	53.0	46.1	52.9	69.9	70.8	18.5	28.4	54.6	**60.7**	67.1	60.4	52.9
WSOD2 [60]	SS	65.1	64.8	**57.2**	**39.2**	24.3	69.8	66.2	61.0	29.8	64.6	42.5	60.1	**71.2**	70.7	21.9	28.1	58.6	59.7	52.2	64.8	53.6
Ours	SS	**68.8**	**77.7**	57.0	27.7	**28.9**	69.1	**74.5**	67.0	**32.1**	**73.2**	48.1	45.2	54.4	**73.7**	**35.0**	29.3	**64.1**	53.8	65.3	**65.2**	**54.9**

Table 9: Single model per-class detection results using VGG16 on PASCAL VOC 2007.

Methods	Proposal	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Motor	Person	Plant	Sheep	Sofa	Train	TV	AP
Fast R-CNN	SS	80.3	74.7	66.9	46.9	37.7	73.9	68.6	87.7	41.7	71.1	51.1	86.0	77.8	79.8	69.8	32.1	65.5	63.8	76.4	61.7	65.7
Faster R-CNN	RPN	82.3	76.4	71.0	48.4	45.2	72.1	72.3	87.3	42.2	73.7	50.0	86.8	78.7	78.4	77.4	34.5	70.1	57.1	77.1	58.9	67.0
Li [27]	EB	62.9	55.5	43.7	14.9	13.6	57.7	52.4	50.9	13.3	45.4	4.0	30.2	55.6	67.0	3.8	23.1	39.4	5.5	50.7	29.3	35.9
ContextLocNet [22]	SS	64.0	54.9	36.4	8.1	12.6	53.1	40.5	28.4	6.6	35.3	**34.4**	49.1	42.6	62.4	**19.8**	15.2	27.0	33.1	33.0	50.0	35.3
OICR [45]	SS	67.7	61.2	41.5	25.6	22.2	54.6	49.7	25.4	19.9	47.0	18.1	26.0	38.9	67.7	2.0	22.6	41.1	34.3	37.9	55.3	37.9
Jie [21]	?	60.8	54.2	34.1	14.9	13.1	54.3	53.4	58.6	3.7	53.1	8.3	43.4	49.8	69.2	4.1	17.5	43.8	25.6	55.0	50.1	38.3
Diba [9]	EB	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	37.9
Shen [37]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	39.1
PCL [44]	SS	58.2	66.0	41.8	24.8	27.2	55.7	55.2	28.5	16.6	51.0	17.5	28.6	49.7	70.5	7.1	25.7	47.5	36.6	44.1	59.2	40.6
Wei [56]	SS	67.4	57.0	37.7	23.7	15.2	56.9	49.1	64.8	15.1	39.4	19.3	48.4	44.5	67.2	2.1	23.3	35.1	40.2	46.6	45.8	40.0
Tang [46]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	40.8
Wan [51]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	42.4
SDCN [28]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	43.5
Yang [59]	SS	64.7	66.3	46.8	28.5	28.4	59.8	58.6	70.9	13.8	55.0	15.7	60.5	**63.9**	69.2	8.7	23.8	44.7	**52.7**	41.5	62.6	46.8
C-MIL [50]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	46.7
WSOD2 [60]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	47.2
Arun [2]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	48.4
C-MIDN [12]	SS	72.9	68.9	53.9	25.3	29.7	60.9	56.0	**78.3**	23.0	57.8	25.7	**73.0**	63.5	73.7	13.1	28.7	51.5	35.0	56.1	57.5	50.2
Ours†	SS	**78.3**	**73.9**	**56.5**	**30.4**	**37.4**	**64.2**	**59.3**	60.3	**26.6**	**66.8**	25.0	55.0	61.8	**79.3**	14.5	**30.3**	**61.5**	40.7	**56.4**	**63.5**	**52.1**

Table 10: Single model per-class detection results using VGG16 on PASCAL VOC 2012.

C.2. Results on ImageNet VID dataset

Additional visualizations of our obtained results on ImageNet VID are shown in Fig. 16, where the frames of the same video are illustrated in the same row. These results are obtained using the ResNet-101 based model. We observe: our model is able to handle objects of different poses, scales, and viewpoints in the videos.

D. Proposal statistics

For consistency with prior literature, we use Selective-Search (SS) [49] for VOC and MCG [1] for COCO. Both methods generate around 2K proposals on average as shown in Tab. 13 but occasionally yield more than 5K on certain images. Our Sequential batch back-propagation can handle these cases easily even with ResNet-101, while other methods quickly run out of memory (Fig. 11 in main paper).

E. Need for redundant proposals

In WSOD, since ground-truth boxes are missing, object proposals have to be redundant for high recall rates, consuming significant amounts of memory. To study the need for a large number of proposals we randomly sample p per-

Methods	Proposal	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Motor	Person	Plant	Sheep	Sofa	Train	TV	CorLoc
Cinbis [7]	SS	56.6	58.3	28.4	20.7	6.8	54.9	69.1	20.8	9.2	50.5	10.2	29.0	58.0	64.9	36.7	18.7	56.5	13.2	54.9	59.4	38.8
Bilen [4]	SS	66.4	59.3	42.7	20.4	21.3	63.4	74.3	59.6	21.1	58.2	14.0	38.5	49.5	60.0	19.8	39.2	41.7	30.1	50.2	44.1	43.7
Wang [52]	SS	80.1	63.9	51.5	14.9	21.0	55.7	74.2	43.5	26.2	53.4	16.3	56.7	58.3	69.5	14.1	38.3	58.8	47.2	49.1	60.9	48.5
Li [27]	EB	78.2	67.1	61.8	38.1	36.1	61.8	78.8	55.2	28.5	68.8	18.5	49.2	64.1	73.5	21.4	47.4	64.6	22.3	60.9	52.3	52.4
WSDDN [5]	EB	65.1	58.8	58.5	33.1	39.8	68.3	60.2	59.6	34.8	64.5	30.5	43.0	56.8	82.4	25.5	41.6	61.5	55.9	65.9	63.7	53.5
Teh [47]	EB	84.0	64.6	70.0	**62.4**	25.8	80.6	73.9	71.5	35.7	81.6	46.5	71.3	79.1	78.8	56.7	34.3	69.8	56.7	77.0	72.7	64.6
ContextLocNet [22]	SS	83.3	68.6	54.7	23.4	18.3	73.6	74.1	54.1	8.6	65.1	47.1	59.5	67.0	83.5	35.3	39.9	67.0	49.7	63.5	65.2	55.1
OICR [45]	SS	81.7	80.4	48.7	49.5	32.8	81.7	85.4	40.1	40.6	79.5	35.7	33.7	60.5	88.8	21.8	57.9	76.3	59.9	75.3	**81.4**	60.6
Jie [21]	?	72.7	55.3	53.0	27.8	35.2	68.6	81.9	60.7	11.6	71.6	29.7	54.3	64.3	88.2	22.2	53.7	72.2	52.6	68.9	75.5	56.1
Diba [9]	EB	83.9	72.8	64.5	44.1	40.1	65.7	82.5	58.9	33.7	72.5	25.6	53.7	67.4	77.4	26.8	49.1	68.1	27.9	64.5	55.7	56.7
Wei [56]	SS	84.2	74.1	61.3	52.1	32.1	76.7	82.9	66.6	42.3	70.6	39.5	57.0	61.2	88.4	9.3	54.6	72.2	60.0	65.0	70.3	61.0
Wan [51]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	61.4
PCL [44]	SS	79.6	85.5	62.2	47.9	37.0	83.8	83.4	43.0	38.3	80.1	50.6	30.9	57.8	90.8	27.0	58.2	75.3	**68.5**	75.7	78.9	62.7
Tang [46]	SS	77.5	81.2	55.3	19.7	44.3	80.2	86.6	69.5	10.1	87.7	**68.4**	52.1	84.4	91.6	**57.4**	**63.4**	77.3	58.1	57.0	53.8	63.8
Li [28]	SS	85.0	83.9	58.9	59.6	43.1	79.7	85.2	77.9	31.3	78.1	50.6	75.6	76.2	88.4	49.7	56.4	73.2	62.6	77.2	79.9	68.6
Shen [37]	SS	82.9	74.0	73.4	47.1	**60.9**	80.4	77.5	**78.8**	18.6	70.0	56.7	67.0	64.5	84.0	47.0	50.1	71.9	57.6	**83.3**	43.5	64.5
C-MIL [50]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	65.0
Yang [59]	SS	80.0	83.9	74.2	53.2	48.5	82.7	86.2	69.5	39.3	82.9	53.6	61.4	72.4	91.2	22.4	57.5	**83.5**	64.8	75.7	77.1	68.0
WSOD2 [60]	SS	87.1	80.0	74.8	60.1	36.6	79.2	83.8	70.6	43.5	**88.4**	46.0	**74.7**	87.4	90.8	44.2	52.4	81.4	61.8	67.7	79.9	69.5
Arun [2]	SS	**88.6**	**86.3**	71.8	53.4	51.2	**87.6**	**89.0**	65.3	33.2	86.6	58.8	65.9	**87.7**	**93.3**	30.9	58.9	83.4	67.8	78.7	80.2	**70.9**
Ours	SS	87.5	82.4	**76.0**	58.0	44.7	82.2	87.5	71.2	**49.1**	81.5	51.7	53.3	71.4	92.8	38.2	52.8	79.4	61.0	78.3	76.0	68.8

Table 11: Single model per-class correct localization (CorLoc) results using VGG16 on PASCAL VOC 2007.

Methods	Proposal	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Motor	Person	Plant	Sheep	Sofa	Train	TV	CorLoc
Li [27]	EB	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	29.1
ContextLocNet [22]	SS	78.3	70.8	52.5	34.7	36.6	80.0	58.7	38.6	27.7	71.2	32.3	48.7	76.2	77.4	16.0	48.4	69.9	47.5	66.9	62.9	54.8
OICR [45]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	62.1
Jie [21]	?	82.4	68.1	54.5	38.9	35.9	84.7	73.1	64.8	17.1	78.3	22.5	57.0	70.8	86.6	18.7	49.7	80.7	45.3	70.1	77.3	58.8
PCL [44]	SS	77.2	83.0	62.1	55.0	49.3	83.0	75.8	37.7	43.2	81.6	46.8	42.9	73.3	90.3	21.4	56.7	84.4	55.0	62.9	82.5	63.2
Wei [56]	SS	79.1	83.9	64.6	50.6	37.8	87.4	74.0	74.1	40.4	80.6	42.6	53.6	66.5	88.8	18.8	54.9	80.4	60.4	70.7	79.3	64.4
Shen [37]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	63.5
Tang [46]	SS	85.5	60.8	62.5	36.6	53.8	82.1	**80.1**	48.2	14.9	87.7	**68.5**	60.7	85.7	89.2	**62.9**	**62.1**	87.1	54.0	45.1	70.6	64.9
Li [28]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	67.9
C-MIL [50]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	67.4
Yang [59]	SS	82.4	83.7	**72.4**	**57.9**	52.9	86.5	78.2	**78.6**	40.1	86.4	37.9	**67.9**	**87.6**	90.5	25.6	53.9	85.0	**71.9**	66.2	84.7	69.5
Arun [2]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	69.5
WSOD2 [60]	SS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	**71.9**
Ours	SS	**91.7**	**85.6**	71.7	56.6	**55.6**	**88.6**	77.3	63.4	**53.6**	**90.0**	51.6	62.6	79.3	**94.2**	32.7	58.8	**90.5**	57.7	**70.9**	**85.7**	70.9

Table 12: Single model per-class correct localization (CorLoc) results using VGG16 on PASCAL VOC 2012.

Data	voc07-train	voc07-val	voc07-test	voc12-train	voc12-val	voc12-test
Avg/Max	2001 / 4663	2001 / 5236	2002 / 5398	2014 / 5254	2010 / 5563	2020/5660
Data	coco14-train	coco14-val	coco17-train	coco17-val	coco-test	-
Avg/Max	1957 / 5143	1958 / 6234	1957 / 6234	1961 / 3774	1947 / 4411	-

Table 13: Proposals statistics.

p	60%	80%	90%	95%	100%
AP	48.4	49.7	50.8	52.1	54.9

Table 14: Effect of using different number of proposals.

cent of all proposals. A VGG16 based model on VOC 2007 is used. The results are summarized in Tab. 14. Reducing the number of proposals even by a small amount significantly reduces accuracy: using 95% of the proposals causes a 2.8% AP drop. This suggests that all proposals should be used for best performance.

F. Additional details on video experiments

In this section, we provide additional details of Sec. 5.4. Following supervised methods for video object detection [63, 58], we experiment on the most popular dataset: ImageNet VID [8]. Frame-level category labels are available during training. For each video, we use the uniformly sampled 15 key-frames from [63] for training. For evaluation, we test on the standard validation set, where per-frame spatial object detection results are evaluated for all the videos.

The two models 'Ours' and 'Ours (MIST only)' are two single-frame baselines with or without Concrete DropBlock (main paper Sec. 4.2). In addition, the memory-efficient sequential batch back-propagation (main paper Sec. 4.3) permits to leverage short-term motion patterns (*i.e.*, optical-flow) to further increase the performance. For 'Ours+flow,' we first use FlowNet2 [19] to compute optical flow between neighboring frames and the reference frame. The estimated flow maps are then used to warp the nearby frames' feature maps to linearly sum with the reference frame for representation enhancement. The accumulated features are then fed into the proposed task head (modules after 'Base' in main paper Fig. 2) for weakly supervised training. This method combines the flow-guided feature warping method as discussed in [63] to leverage temporal coherence and the proposed WSOD task head to handle frame-level weak supervision. Hence it achieves better results than the aforementioned two baselines ('Ours' and 'Ours (MIST only)') using both VGG16 and ResNet-101 as reported in Tab. 7.

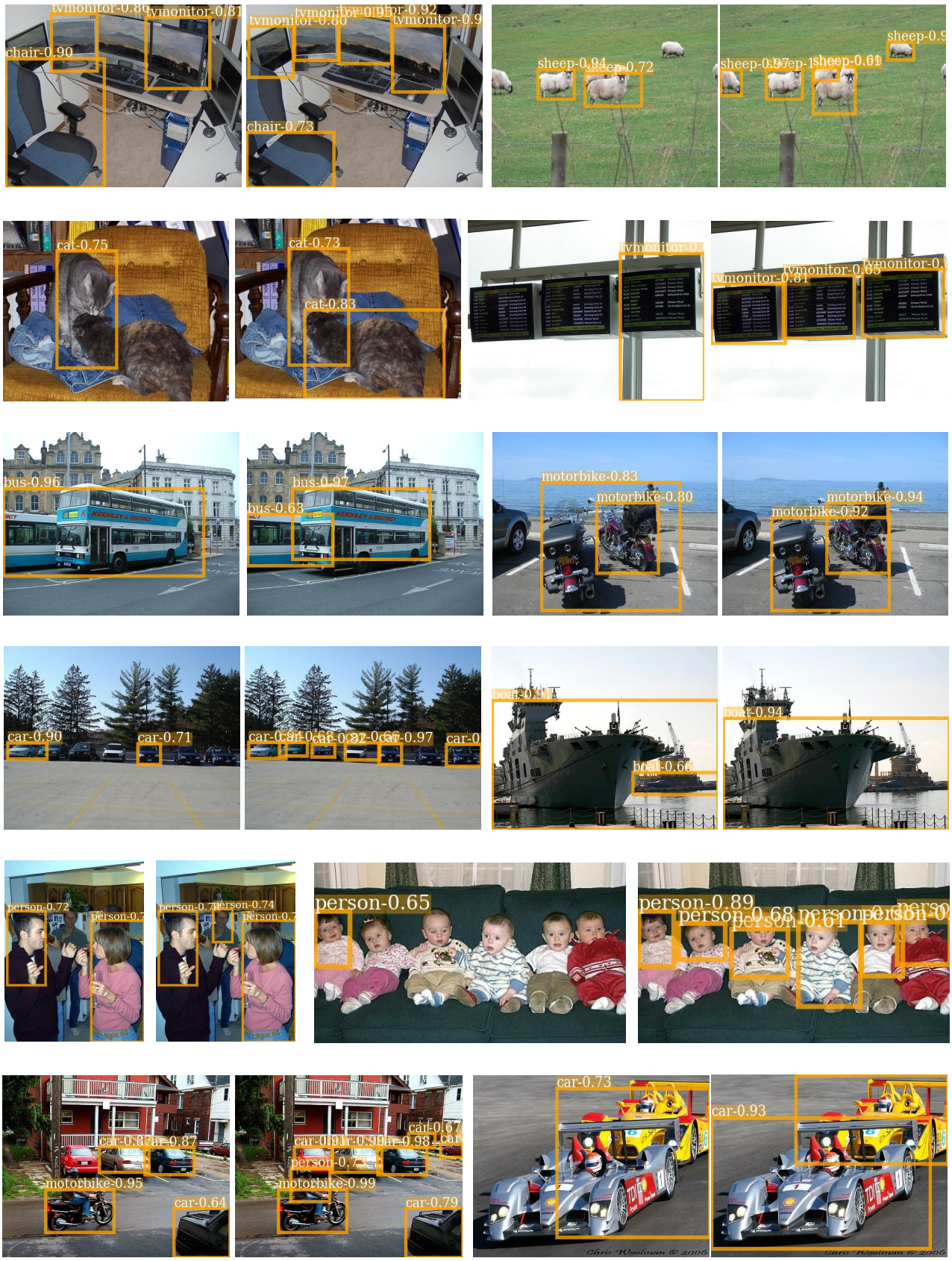

Figure 13: Examples that highlight cases of 'Instance Ambiguity'. For every pair: baseline (left) and our model (right).

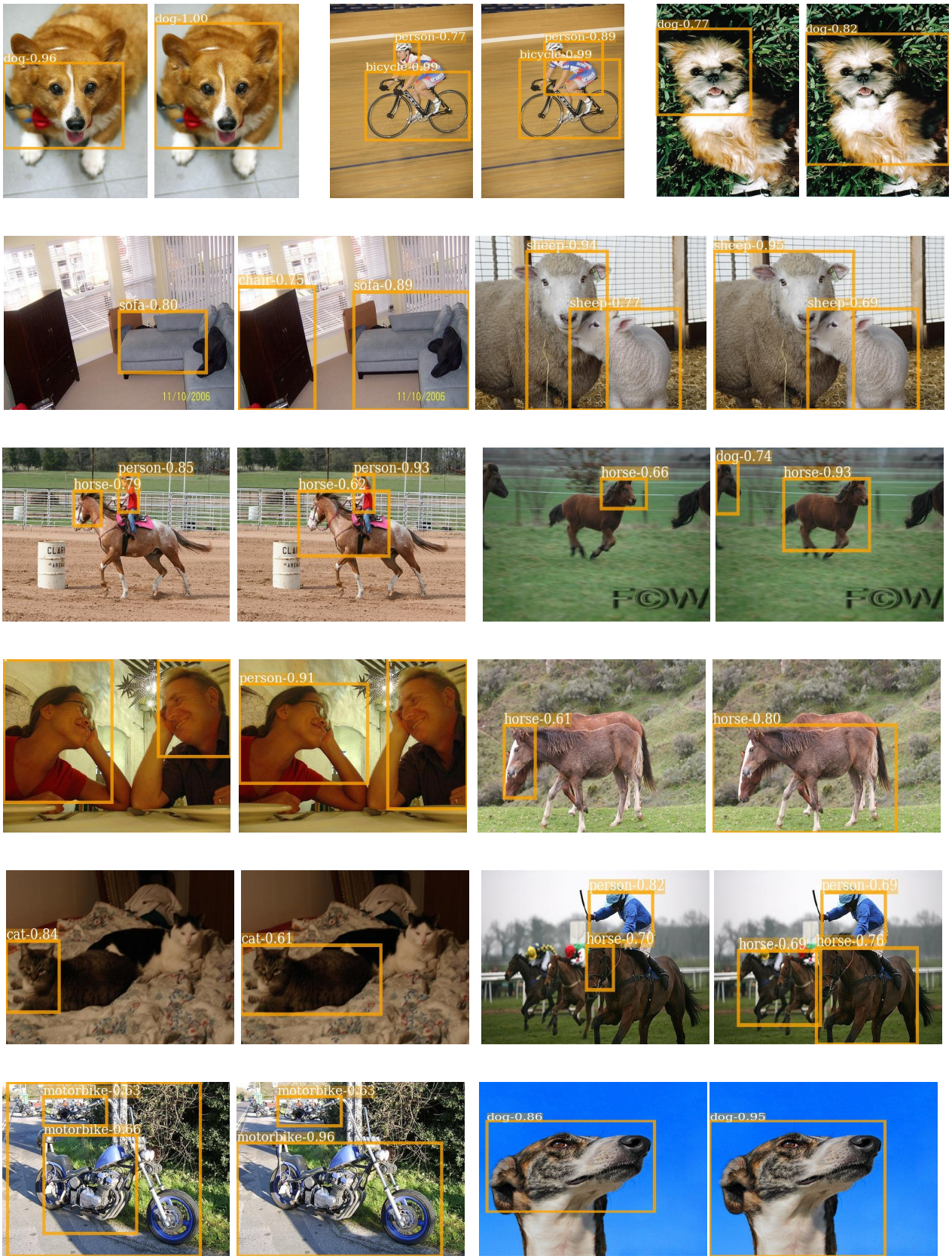

Figure 14: Examples that highlight cases of 'Part Domination'. For every pair: baseline (left) and our model (right).

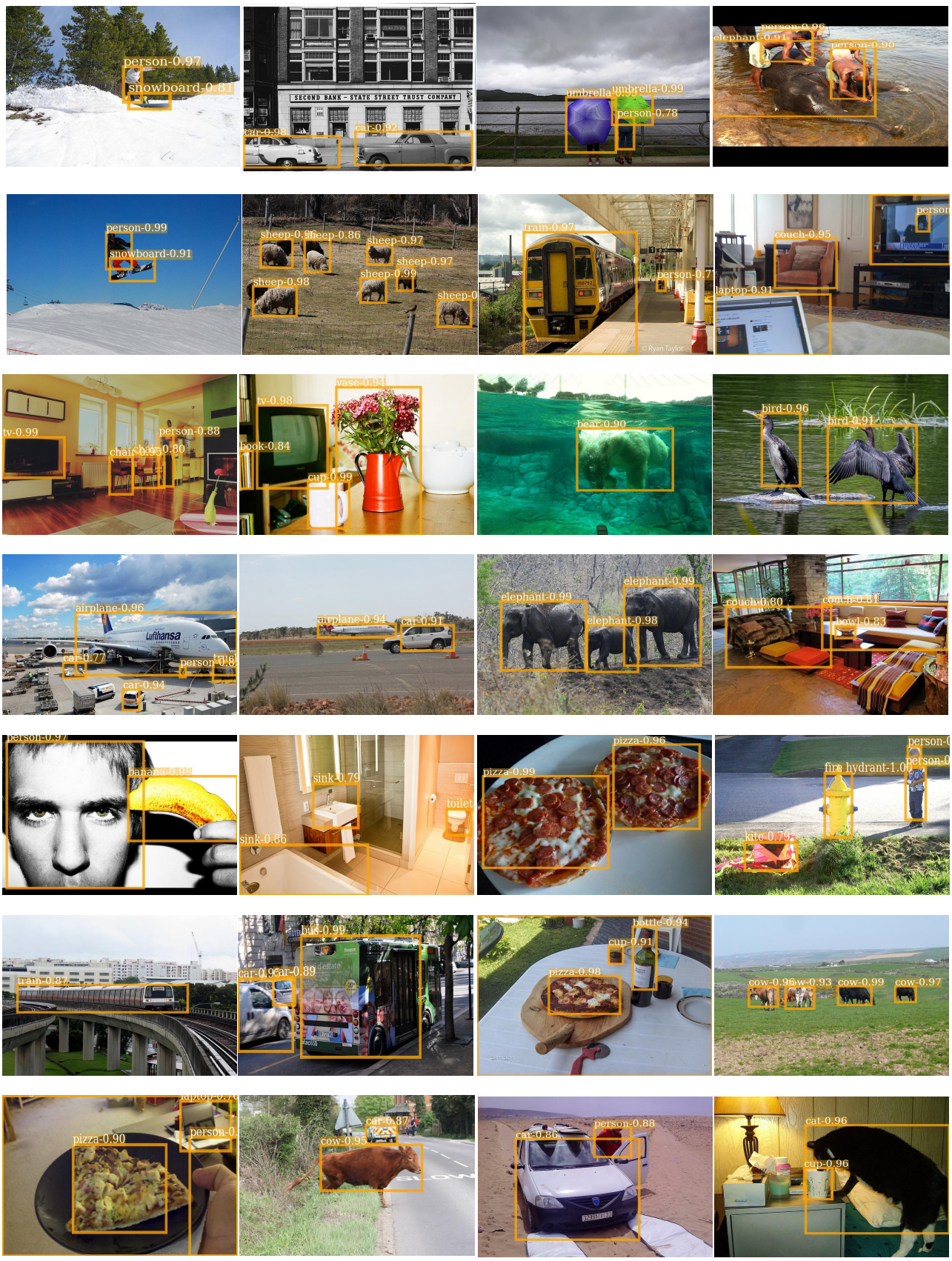

Figure 15: Additional visualization results of the proposed method on the COCO2014 validation set.

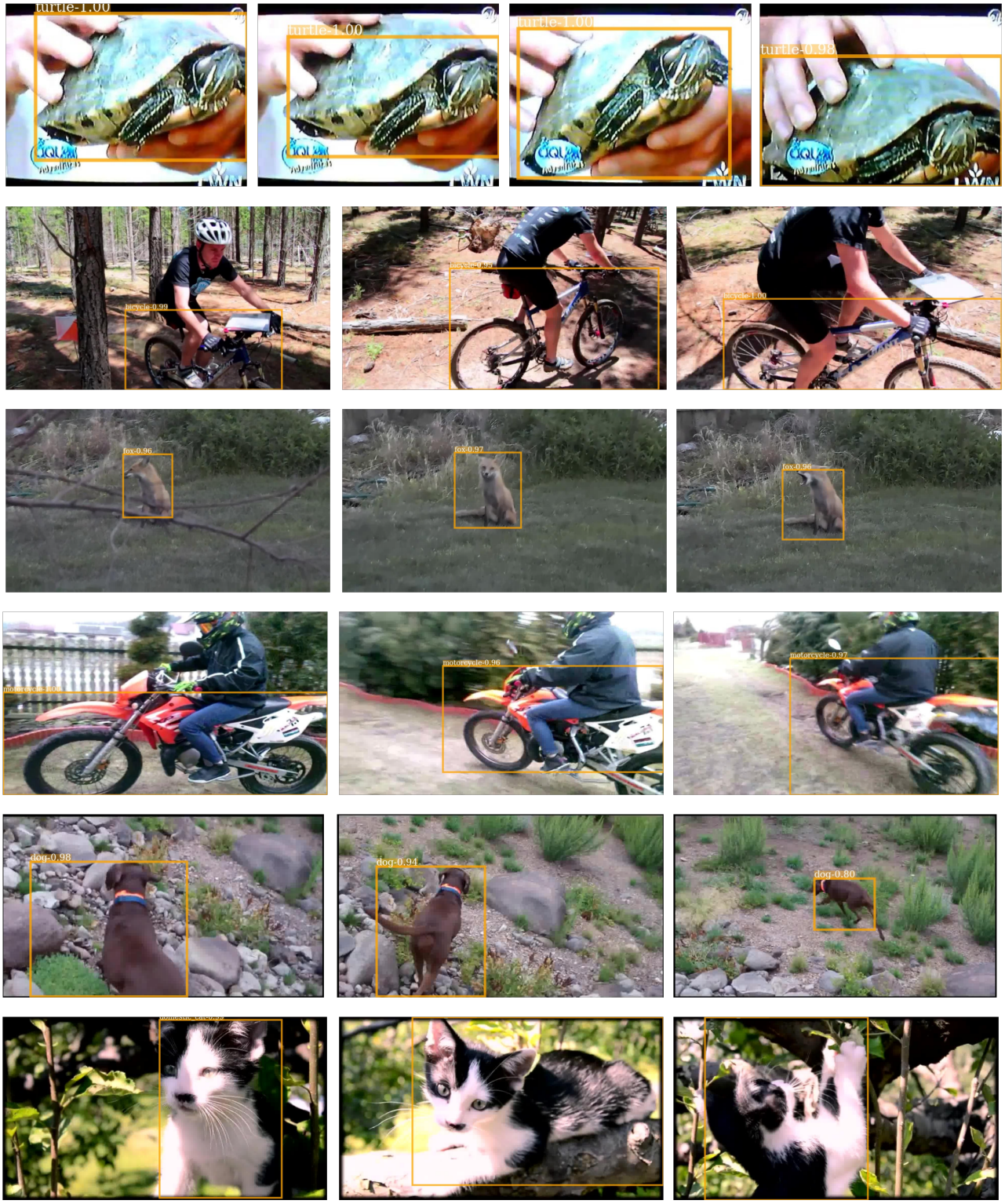

Figure 16: Additional visualization results of the proposed method on the ImageNet VID validation set.