Regional Semantic Contrast and Aggregation for Weakly Supervised Semantic Segmentation
Supplemental Material

Tianfei Zhou[1,*], Meijie Zhang[2,*], Fang Zhao[3], Jianwu Li[2,†]
[1] Computer Vision Lab, ETH Zurich [2] Beijing Institute of Technology [3] Inception Institute of AI
https://github.com/maeve07/RCA.git

In this document, we provide additional contents including pseudo codes of essential modules (§S1), per-class quantitative segmentation results on VOC 2012 `val` and `test` (§S2), more visualization results on VOC 2012 [5] and COCO 2014 [11] (§S3), as well as limitation (§S4) and societal impact analysis (§S5).

S1. Pseudo Code

Algorithms 1 and 2 provide the pseudo-codes of regional semantic contrast and regional semantic aggregation, respectively. In addition, we give a pseudo-code of the model inference procedure in Algorithm 3.

S2. Per-Class Result on VOC 2012

Table S1 and Table S2 list per-class segmentation scores on VOC 2012 `val` and `test`, respectively. We observe that RCA obtains the best performance on most of the categories (*e.g.*, `aeroplane`, `car`, `motorcycle`). These detailed results further confirm the effectiveness of our approach.

S3. Additional Visualization Result

S3.1. Object Localization Result

Fig. S1 depicts more object localization results on VOC 2012 `train`. We observe that RCA yields a clearly perceivable improvement over the OAA^{++} baseline. In particular, RCA is able to reveal the full extents of objects, even for too small or too large ones. In multi-object scenarios, it can accurately identify all objects, while OAA^{++} only provides sparse responses for some of them.

S3.2. Semantic Segmentation Result

In Fig. S2 and Fig. S3, we show extra segmentation results of RCA on VOC 2012 `val` and COCO 2014 `val`, respectively. Consistent with our analysis in the main paper, we observe that RCA is able to produce accurate segmentation results with crisp boundaries in diverse scenarios.

Algorithm 1 PyTorch-style pseudocode of regional semantic contrast (RSC).

```
# F: image feature (W x H x D, Eqn.(1))
# M = {M_1, ..., M_L}: memory bank
# beta: shape parameter
# t: temperature

def RSC(F, y):
    losses = []
    for l in range(1, L):
        if y_l == 0:
            continue

        # region feature: D-dimensional, Eqn.(2)
        f_l = MAP(F, P_l)

        loss = loss_rm_nce(f_l, y_l)
        losses.append(loss)
    return losses.mean()


# Region Mixup InfoNCE loss in Eqn.(6)
def loss_rm_nce(f_l, y_l):
    # sample another region f_l_neg such that
    # y_l_neg != y_l
    f_l_neg = mixup_sampling()

    # beta sampling
    omega = Beta(beta, beta)

    # region mixup
    f_l_hat = omega * f_l + (1 - omega) * f_l_neg
    f_l_hat = l2_norm(f_l_hat)

    # region mixup InfoNCE
    loss = omega * loss_nce(f_l_hat, y_l) + (1 - omega
        ) * loss_nce(f_l_hat, y_l_neg)

    return loss

# InfoNCE loss in Eqn.(4)
def loss_nce(f_l, y_l):
    # m_l_pos: postive features from M_l
    # m_l_neg: negative features from M\M_l

    logits_neg = torch.einsum('d,nd->n', [f_l, m_l_neg
        ]) / t
    logits_pos = torch.einsum('d,nd->n', [f_l, m_l_pos
        ]) / t

    loss = (torch.log(logits_neg + torch.exp(
        logits_pos)) - logits_pos).mean()

    return loss
```

`mm`: matrix multiplication; `cat`: concatenation; `l2_norm`: ℓ_2 normalization; `mixup_sampler`: a random sampler to find another region for region mixup; `Beta`: Beta distribution; `MAP`: masked average pooling.

[*] Equal contributions; [†] Corresponding author: Jianwu Li.

Algorithm 2 PyTorch-style pseudocode of regional semantic aggregation (RSA).

```
# F: image feature (W x H x D, Eqn.(1))
# Q: prototypical representation (LK x D)

def RSA(F, Q):
    # affinity: WH x LK, Eqn.(7)
    S = softmax(mm(F, Q.transpose()))

    # context summary: WH x D, Eqn.(8)
    F_prime = mm(S, Q)

    # feature concatenation: W x H x 2D, Eqn.(9)
    F_hat = cat([F, F_prime], dim=-1)

    return F_hat
```

`mm`: matrix multiplication; `cat`: concatenation.

Algorithm 3 PyTorch-style pseudocode of model inference.

```
# I: test image
# Q: prototypical representation (LK x D)
# FCN: backbone network
# CAM: class-aware convolutional layer

# image feature: W x H x D, Eqn.(1)
F = FCN(I)

F_hat = RSA(F, Q)

# CAM prediction: W x H x L, Eqn.(10)
O = CAM(F_hat)
```

S4. Limitation

One downside of RCA is that it needs to maintain an external memory bank during training, thereby increasing the memory complexity. However, we show in the main paper that RCA is not sensitive with the memory size, and we can use a small size (*e.g.*, 500) to achieve comparable performance.

S5. Potential Societal Impact

RCA shows high potential impact for many practical applications, *e.g.*, autonomous driving, medical imaging, and transportation, where expert annotation is expensive. However, the model can be deployed in human monitoring and surveillance as well which raise ethical and privacy issues. It can be avoided by enforcing a strict and secure data privacy regulation to regulate the technology.

References

[1] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018. S3

[2] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *CVPR*, 2020. S3

[3] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *CVPR*, 2020. S3

[4] Liyi Chen, Weiwei Wu, Chenchen Fu, Xiao Han, and Yuntao Zhang. Weakly supervised semantic segmentation with boundary exploration. In *ECCV*, 2020. S3

[5] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. S1, S3, S4, S5

[6] Junsong Fan, Zhaoxiang Zhang, Tienniu Tan, Chunfeng Song, and Jun Xiao. Cian: Cross-image affinity net for weakly supervised semantic segmentation. In *AAAI*, 2020. S3

[7] Qibin Hou, PengTao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. In *NeurIPS*, 2018. S3

[8] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *CVPR*, 2018. S3

[9] Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *ECCV*, 2016. S3

[10] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *CVPR*, 2019. S3

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. S1, S6

[12] Xiang Wang, Shaodi You, Xi Li, and Huimin Ma. Weakly-supervised semantic segmentation by iteratively mining common object features. In *CVPR*, 2018. S3

[13] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *CVPR*, 2020. S3

[14] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, Ferdous Sohel, and Dan Xu. Leveraging auxiliary tasks with affinity learning for weakly supervised semantic segmentation. In *ICCV*, 2021. S3

[15] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, and Lihe Zhang. Joint learning of saliency detection and weakly supervised semantic segmentation. In *ICCV*, 2019. S3

[16] Bingfeng Zhang, Jimin Xiao, Yunchao Wei, Mingjie Sun, and Kaizhu Huang. Reliability does matter: An end-to-end weakly supervised semantic segmentation approach. In *AAAI*, 2020. S3

[17] Tianyi Zhang, Guosheng Lin, Weide Liu, Jianfei Cai, and Alex Kot. Splitting vs. merging: Mining object regions with discrepancy and intersection loss for weakly supervised semantic segmentation. In *ECCV*, 2020. S3

[18] Tianfei Zhou, Liulei Li, Xueyi Li, Chun-Mei Feng, Jianwu Li, and Ling Shao. Group-wise learning for weakly supervised semantic segmentation. *IEEE TIP*, 31:799–811, 2021. S3

method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mIoU (%)
SEC [9]	82.4	62.9	26.4	61.6	27.6	38.1	66.6	62.7	75.2	22.1	53.5	28.3	65.8	57.8	62.3	52.5	32.5	62.6	32.1	45.4	45.3	50.7
MCOF [12]	87.0	78.4	29.4	68.0	44.0	67.3	80.3	74.1	82.2	21.1	70.7	28.2	73.2	71.5	67.2	53.0	47.7	74.5	32.4	71.0	45.8	60.3
AffinityNet [1]	88.2	68.2	30.6	81.1	49.6	61.0	77.8	66.1	75.1	29.0	66.0	40.2	80.4	62.0	70.4	73.7	42.5	70.7	42.6	68.1	51.6	61.7
SeeNet [7]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	63.1
FickleNet [10]	89.5	76.6	32.6	74.6	51.5	71.1	83.4	74.4	83.6	24.1	73.4	47.4	78.2	74.0	68.8	73.2	47.8	79.9	37.0	57.3	**64.6**	64.9
SSNet [15]	90.0	77.4	37.5	80.7	61.6	67.9	81.8	69.0	83.7	13.6	79.4	23.3	78.0	75.3	71.4	68.1	35.2	78.2	32.5	75.5	48.0	63.3
CIAN [6]	88.2	79.5	32.6	75.7	56.8	72.1	85.3	72.9	81.7	27.6	73.3	39.8	76.4	77.0	74.9	66.8	46.6	81.0	29.1	60.4	53.3	64.3
RRM [16]	87.9	75.9	31.7	78.3	54.6	62.2	80.5	73.7	71.2	30.5	67.4	40.9	71.8	66.2	70.3	72.6	49.0	70.7	38.4	62.7	58.4	62.6
SubCat [3]	88.8	51.6	30.3	82.9	53.0	75.8	88.6	74.8	86.6	**32.4**	79.9	**53.8**	82.3	78.5	70.4	71.2	40.2	78.3	**42.9**	66.8	58.8	66.1
SS-WSSS [2]	88.7	70.4	35.1	75.7	51.9	65.8	71.9	64.2	81.1	30.8	73.3	28.1	81.6	69.1	62.6	74.8	48.6	71.0	40.1	68.5	64.3	62.7
SEAM [13]	88.8	68.5	33.3	**85.7**	40.4	67.3	78.9	76.3	81.9	29.1	75.5	48.1	79.9	73.8	71.4	75.2	48.9	79.8	40.9	58.2	53.0	64.5
Zhang *et al* [17]	90.4	85.6	38.9	78.9	62.0	73.4	83.7	74.3	82.9	25.8	77.8	30.1	81.1	79.3	76.1	73.9	38.6	85.0	32.7	72.8	55.7	66.6
BES [4]	88.9	74.1	29.8	81.3	53.3	69.9	**89.4**	79.8	84.2	27.9	76.9	46.6	78.8	75.9	72.2	70.4	**50.8**	79.4	39.9	65.3	44.8	65.7
GroupWSSS [18]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	68.7
AuxSegNet [14]	91.7	82.5	38.2	84.3	67.4	**76.7**	85.0	79.8	**90.7**	24.5	81.2	22.7	**86.7**	78.7	76.0	**82.2**	37.9	**86.4**	39.3	**75.6**	61.0	69.0
RCA	**91.8**	**88.4**	**39.1**	85.1	**69.0**	75.7	86.6	**82.3**	89.1	28.1	**81.9**	37.9	85.9	**79.4**	**82.1**	78.6	47.7	84.4	34.9	75.4	58.6	**70.6**

Table S1. **Per-class segmentation results** on VOC 2012 [5] `val`. See §S2 for details.

method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mIoU (%)
DSRG [8]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	63.2
MCOF [12]	88.2	80.8	31.4	70.9	34.9	65.7	83.5	75.1	79.0	22.0	70.3	31.7	77.7	72.9	77.1	56.9	41.8	74.9	36.6	71.2	42.6	61.2
AffinityNet [1]	89.1	70.6	31.6	77.2	42.2	68.9	79.1	66.5	74.9	29.6	68.7	56.1	82.1	64.8	78.6	73.5	50.8	70.7	47.7	63.9	51.1	63.7
FickleNet [10]	89.8	78.3	34.1	73.4	41.2	67.2	81.0	77.3	81.2	29.1	72.4	47.2	76.8	76.5	76.1	72.9	56.5	82.9	43.6	48.7	**64.7**	65.3
SSNet [15]	90.4	85.4	37.9	77.2	48.2	64.5	83.9	74.8	83.4	15.9	72.4	34.3	80.0	77.3	78.5	69.0	41.9	76.3	38.3	72.3	48.2	64.3
RRM [16]	87.8	77.5	30.8	71.7	36.0	64.2	75.3	70.4	81.7	29.3	70.4	52.0	78.6	73.8	74.4	72.1	54.2	75.2	50.6	42.0	52.5	62.9
SS-WSSS [2]	89.2	73.4	37.3	68.3	45.8	68.0	72.7	64.1	74.1	**32.9**	74.9	39.2	81.3	74.6	72.6	75.4	58.1	71.0	48.7	67.7	60.1	64.3
SEAM [13]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	65.7
BES [4]	89.0	72.7	30.4	84.6	47.5	63.0	86.8	80.7	85.2	30.1	76.5	**56.4**	81.8	79.9	77.0	67.8	48.6	82.3	**57.2**	54.0	46.7	66.6
AuxSegNet [14]	91.6	85.1	39.4	80.0	51.4	**69.9**	81.4	79.9	86.5	26.6	75.3	29.7	81.7	83.6	78.0	**83.1**	56.1	84.5	39.8	**77.2**	60.9	68.6
GroupWSSS [18]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	69.0
RCA	**92.1**	**86.6**	**40.0**	**90.1**	**60.4**	68.2	**89.8**	**82.3**	**87.0**	27.2	**86.4**	32.0	**85.3**	**88.1**	**83.2**	78.0	**59.2**	**86.7**	45.0	71.3	52.5	**71.0**

Table S2. **Per-class segmentation results** on VOC 2012 [5] `test`. See §S2 for details.

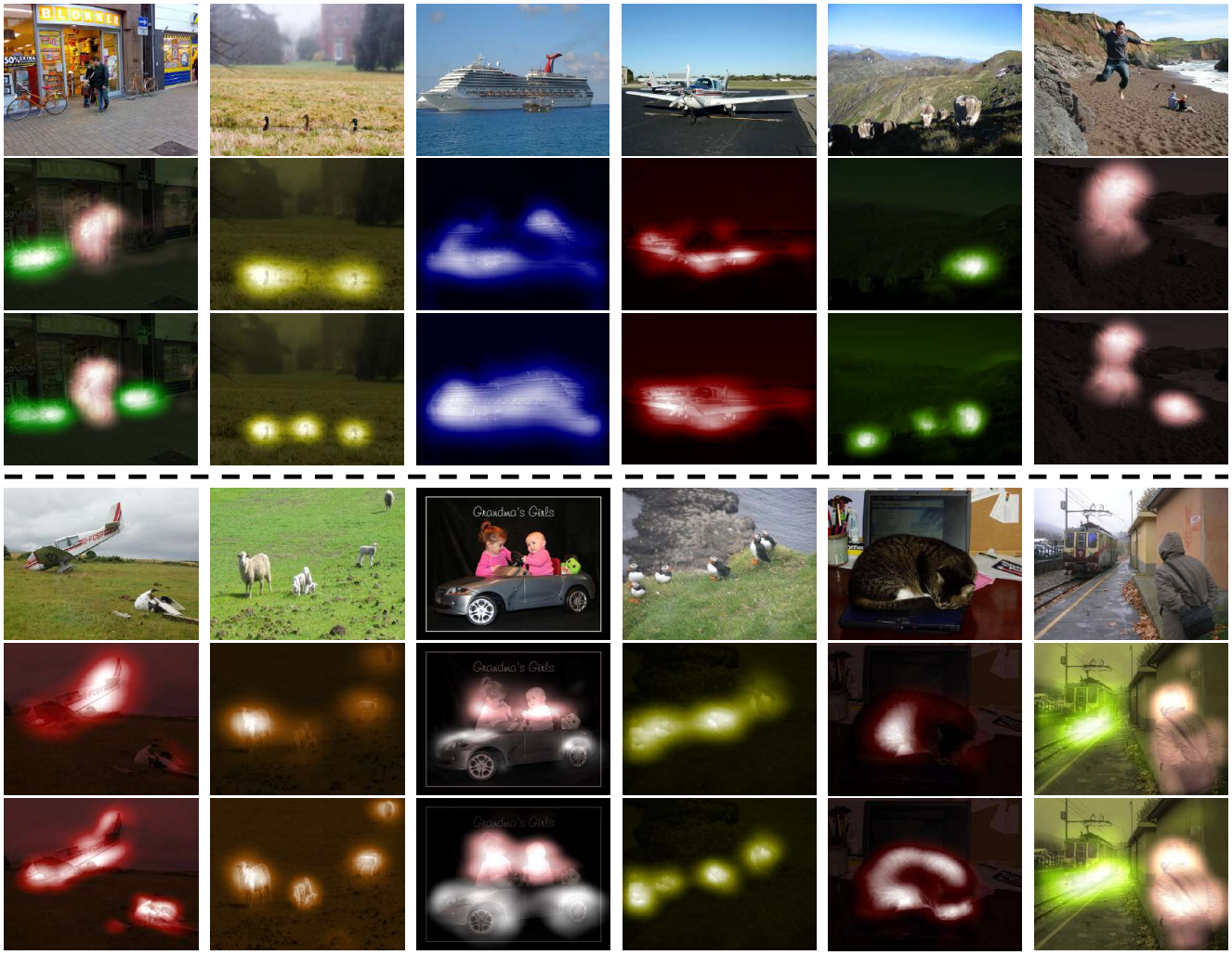

Figure S1. **Visualization of class activation maps** on VOC 2012 [5] `train`. From top to bottom: input images, results of OAA^{++}, results of RCA. See §S3 for details.

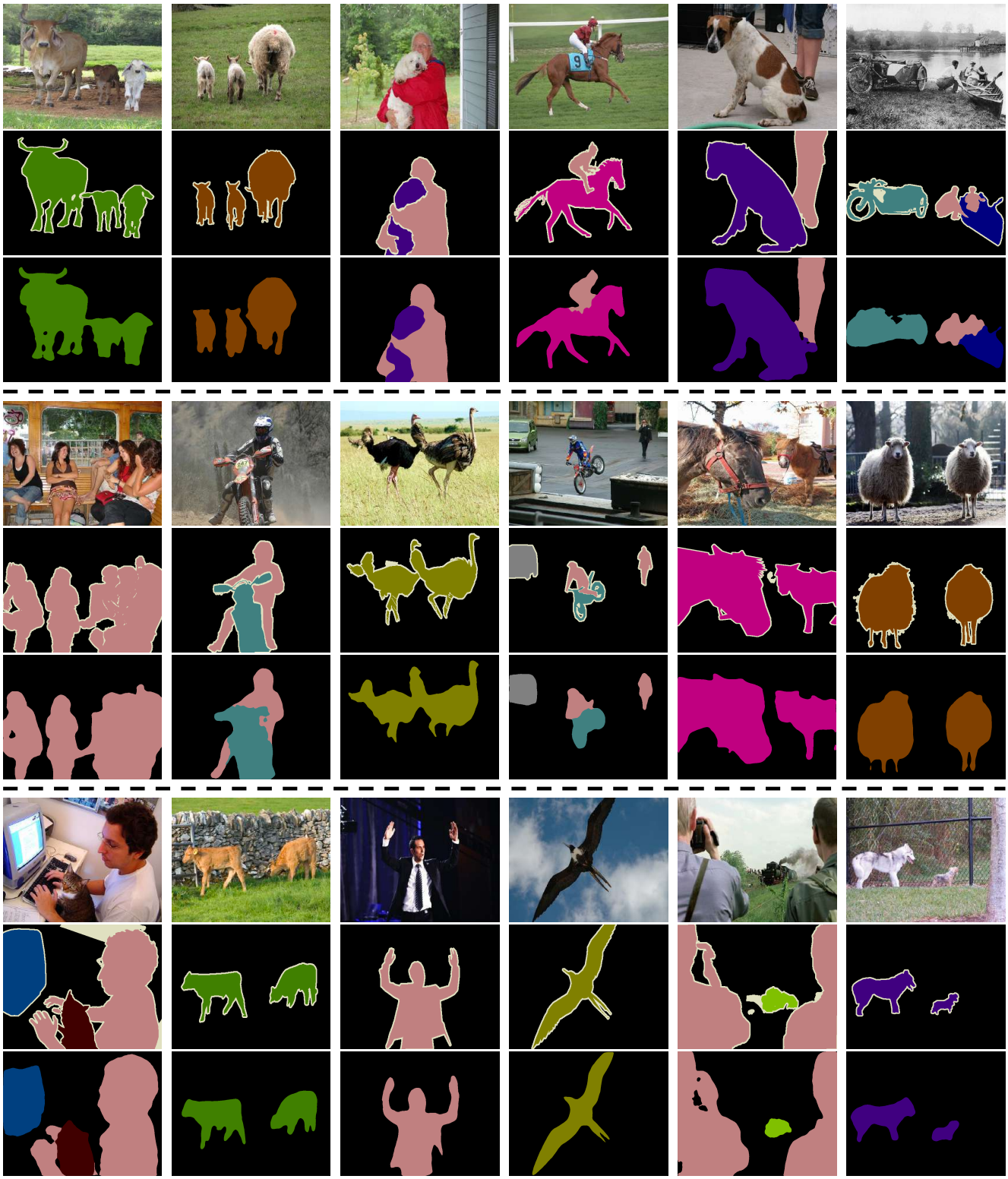

Figure S2. **Qualitative segmentation results** on VOC 2012 [5] `val`. From top to bottom: input images, ground-truths, our results. See §S3 for details.

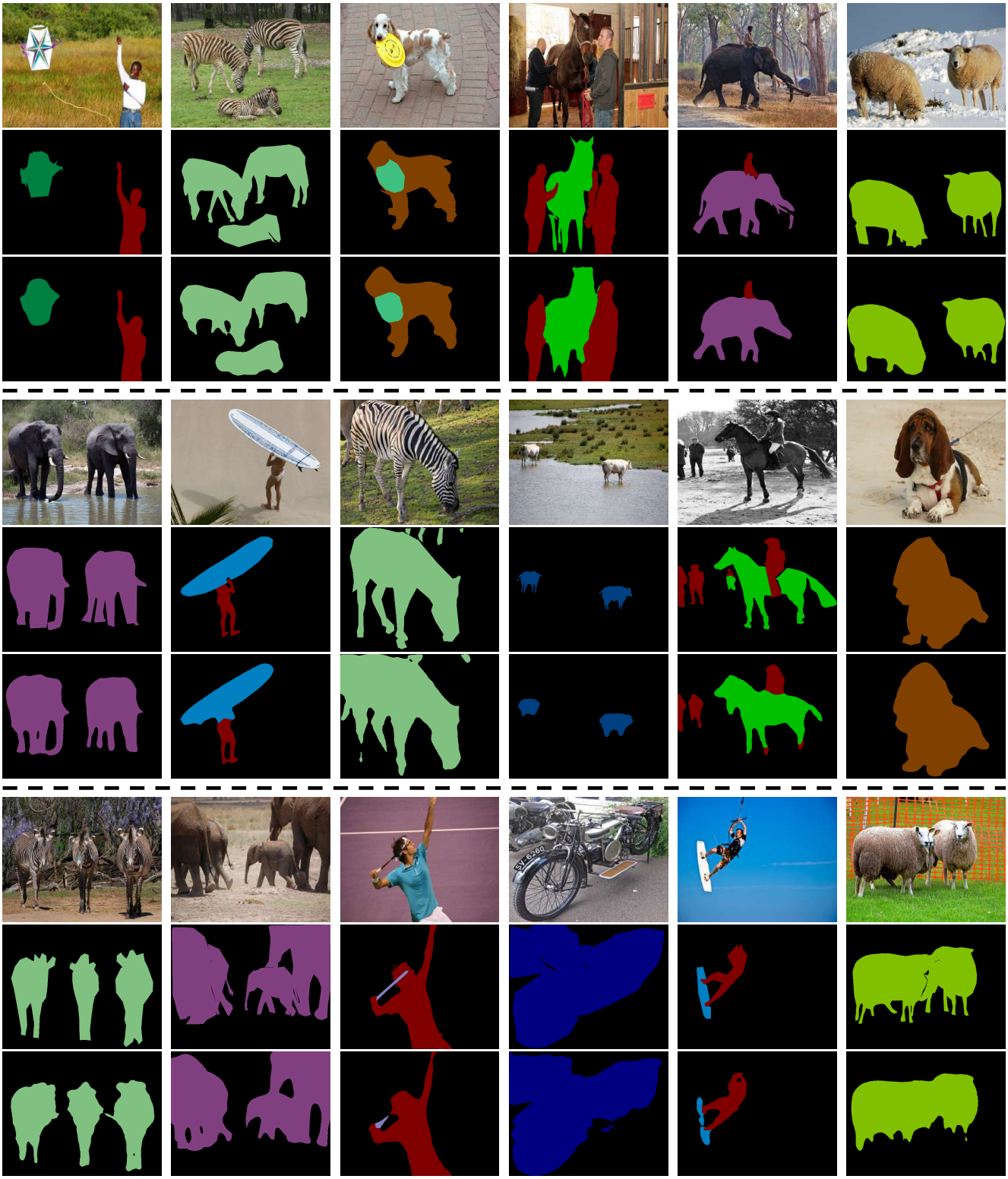

Figure S3. **Qualitative segmentation results** on COCO 2014 [11] `val`. From top to bottom: input images, ground-truths, our results. See §S3 for details.