Improving Semantic Segmentation via Decoupled Body and Edge Supervision Supplementary

Xiangtai Li[1] *, Xia Li[1,4], Li Zhang[2], Guangliang Cheng[3] **, Jianping Shi[3], Zhouchen Lin[1], Shaohua Tan[1], and Yunhai Tong[1] **

[1] Key Laboratory of Machine Perception, MOE, School of EECS, Peking University
[2] Department of Engineering Science, University of Oxford
[3] SenseTime Research
[4] Zhejiang Lab

We will give more detailed results on Cityscapes in this supplemental material.

1 More Experimental Details

More implementation details on the edge supervision: As discussed in the paper, we use the edge supervision for two purposes: one for edge prediction and one for edge hardest pixel mining. For edge prediction, we use binary edge map generated from the category label map as the ground-truth, and add extra weights by counting the reciprocal of positive and negative pixels to balance the binary cross-entropy loss. For edge hardest pixel mining, we set $K = 0.1$ based on the input resolution due to the various sizes in different datasets. Since our method requires a fine-grained edge from the ground truth mask, we only use the fine-annotated set.

More implementation details on related methods: For SPN [1], we use the authors' original Pytorch code [1] and append it after the FCN output to replace our proposed Body Generation module. For DCN [2], we use the implementation of mmdetction repo [3] and replace our Body Generation module with two DCN operators. Note that both supervisions and the edge preservation module are kept untouched. For G-SCNN [4], we port the author's open sourced code [4] into our framework with extra shape stream and dual-task loss on the FCN.
Ablation study on component design in body generation (BG). Here we carry out more detailed explorations on our BG design. We adopt the same setting shown in the experiment part. Table 1(a) shows that depth-wise conv works better than bilinear and is also slightly better than naive conv with less computation. Table 1(b) shows that naive bilinear upsampling works better than deconvolution and nearest neighbor during upsampling. Table 1(c) shows that two successive strided-conv with total stride 4 works the best while larger stride leads to degradation due to the loss of details.

* Work done while at SenseTime. Email: lxtpku@pku.edu.cn
** Correspond to: chengguangliang@sensetime.com, yhtong@pku.edu.cn

Method	mIoU (%)	Δ(%)
FCN (Baseline)	76.6	
bilinear	78.2	1.6↑
naive conv	79.7	3.1↑
depth-wise-conv	**80.1**	**3.5**↑

(a) Ablation study on downsampling operations in BG.

Method	mIoU (%)	Δ(%)
FCN (Baseline)	76.6	-
de-conv	79.0	2.4↑
nearest neighbor	79.5	2.9↑
bilinear	**80.1**	**3.5**↑

(b) Ablation study on upsampling operations in BG.

Method	mIoU (%)	Δ(%)
FCN (Baseline)	76.6	-
(1, stride=2)	79.2	2.6↑
(2, stride=4)	**80.1**	**3.5**↑
(3, stride=8)	78.8	2.2↑
(4, stride=16)	78.5	1.9↑

(c) Ablation study on number of strided-convs in BG.

Table 1. Experiment results on Cityscapes validation set with component design in body generation part.

Method	Thrs	mIoU	road	swalk	build.	wall	fence	pole	tlight	sign	veg	terrain	sky	person	rider	car	truck	bus	train	motor	bike
Deeplabv3+	3px	69.7	83.7	65.1	69.7	52.2	46.2	72.2	62.8	67.7	71.8	52.2	80.9	61.5	66.4	78.8	78.2	83.9	91.7	77.9	60.9
G-SCNN	3px	73.6	85.0	68.8	74.1	53.3	47.0	79.6	74.3	76.2	75.3	53.1	83.5	69.8	73.1	83.4	75.8	88.0	93.9	75.1	68.5
Ours	3px	**73.8**	85.2	69.1	74.0	50.3	50.2	78.6	74.6	75.2	75.1	55.3	81.5	70.2	72.1	82.4	76.3	89.1	92.8	76.2	69.5
Deeplabv3+	5px	74.7	88.1	72.6	78.1	55.0	49.1	77.9	69.0	74.7	81.0	55.8	86.4	69.0	71.9	85.4	79.4	85.4	92.1	79.4	68.4
G-SCNN	5px	77.6	88.7	75.3	80.9	55.9	49.9	83.6	78.6	80.4	83.4	56.6	88.4	75.4	77.8	88.3	77.0	88.9	94.2	76.9	75.1
Ours	5px	**79.2**	88.6	74.6	81.8	55.2	55.3	83.3	80.0	80.6	82.9	60.3	88.2	75.4	79.5	89.2	83.6	92.8	96.3	80.9	75.5
Deeplabv3+	9px	78.7	91.2	78.3	84.8	58.1	52.4	82.1	73.7	79.5	87.9	59.4	89.5	74.7	76.8	90.0	80.5	86.6	92.5	81.0	75.4
G-SCNN	9px	80.7	91.3	80.1	86.0	58.5	52.9	86.1	81.5	83.3	89.0	59.8	91.1	79.1	81.5	91.5	78.1	89.7	94.4	78.5	80.4
Ours	9px	**82.3**	91.5	79.7	87.4	57.7	58.3	86.1	83.1	83.8	88.9	63.7	90.8	79.3	83.5	92.5	84.6	93.5	96.6	82.4	82.4
Deeplabv3+	12px	80.1	92.3	80.4	87.2	59.6	53.7	83.8	75.2	81.2	90.2	60.8	90.4	76.6	78.7	91.6	81.0	87.1	92.6	81.8	78.0
G-SCNN	12px	81.8	92.2	81.7	87.9	59.6	54.3	87.1	82.3	84.4	90.9	61.1	91.9	80.4	82.8	92.6	78.5	90.0	94.6	79.1	82.2
Ours	12px	**83.5**	92.4	81.5	89.4	58.8	59.5	87.1	83.9	84.9	91.0	65.0	91.6	80.6	84.9	93.5	85.1	93.7	96.7	82.9	83.1

Table 2. Per-category F-score results on the Cityscapes val set for 4 different thresholds based on Deeplabv3+. Note that our methods output G-SCNN over **all four thresholds**. Best view on screen and zoom in.

Detailed boundaries improvements analysis: We analyze the improvements over boundaries using F-score [5] on the Cityscapes dataset. The analysis is performed on Deeplabv3+ over each class with 4 different boundary thresholds. For a fair comparison, we also include the G-SCNN [4] results on the val set since both models are based on Deeplabv3+. The results are shown in Tab. 2.

Detailed results on the Cityscapes test server: We first give the comparison results with models trained with only fine-annotated data using ResNet-101 as the backbone in Tab. 3. Our method leads to a significant margin with previous state-of-the-art models and outperforms them in 18 of 19 categories. Then we compare the our model with Wider-ResNet in Tab. 4. For both cases, we achieve state-of-the-art results.

2 More Visualisation Results

In this section, we give more visualization examples, as shown in the paper's Experiment parts.

Method	mIoU	road	swalk	build.	wall	fence	pole	tlight	sign	veg	terrain	sky	person	rider	car	truck	bus	train	motor	bike
PSPNet [6]	78.4	98.6	86.2	92.9	50.8	58.8	64.0	75.6	79.0	93.4	72.3	95.4	86.5	71.3	95.9	68.2	79.5	73.8	69.5	77.2
AAF [7]	79.1	98.5	85.6	93.0	53.8	58.9	65.9	75.0	78.4	93.7	72.4	95.6	86.4	70.5	95.9	73.9	82.7	76.9	68.7	76.4
DenseASPP [8]	80.6	**98.7**	87.1	93.4	60.7	62.7	65.6	74.6	78.5	93.6	72.5	95.4	86.2	71.9	96.0	78.0	90.3	80.7	69.7	76.8
DANet [9]	81.5	98.6	87.1	93.5	56.1	**63.3**	69.7	77.3	81.3	93.9	72.9	95.7	87.3	72.9	96.2	76.8	89.4	86.5	72.2	78.2
Ours	**82.8**	**98.7**	**87.2**	**93.9**	**62.1**	62.9	**71.2**	**78.5**	**81.8**	**94.0**	**73.3**	**96.0**	**88.1**	**74.4**	**96.5**	**79.4**	**92.5**	**89.8**	**73.3**	**78.7**

Table 3. Per-category results on the Cityscapes test set. Note that all the models are trained with **only fine annotated data**. Our method outperforms existing approaches on **18** out of 19 categories, and achieves **82.8**% in mIoU.

Method	Coarse	mIoU	road	swalk	build.	wall	fence	pole	tlight	sign	veg	terrain	sky	person	rider	car	truck	bus	train	motor	bike
PSP-Net [6]	✓	81.2	98.7	86.9	93.5	58.4	63.7	67.7	76.1	80.5	93.6	72.2	95.3	86.8	71.9	96.2	77.7	91.5	83.6	70.8	77.5
DeepLabV3 [10]	✓	81.3	98.6	86.2	93.5	55.2	63.2	70.0	77.1	81.3	93.8	72.3	95.9	87.6	73.4	96.3	75.1	90.4	85.1	72.1	78.3
DeepLabV3+ [11]	✓	81.9	98.7	87.0	93.9	59.5	63.7	71.4	78.2	82.2	94.0	73.0	95.8	88.0	73.3	96.4	78.0	90.9	83.9	73.8	78.9
AutoDeepLab-L [12]	✓	82.1	**98.8**	87.6	93.8	61.4	64.4	71.2	77.6	80.9	94.1	72.7	96.0	87.8	72.8	96.5	78.2	90.9	88.4	69.0	77.6
DPC [13]	✓	82.7	98.7	87.1	93.8	57.7	63.5	71.0	78.0	82.1	94.0	73.3	95.4	88.2	**74.5**	96.5	**81.2**	93.3	89.0	74.1	79.0
G-SCNN [4]	×	82.8	98.7	87.4	94.2	61.9	**64.6**	**72.9**	**79.6**	82.5	**94.3**	**74.3**	**96.2**	88.3	74.2	96.0	77.2	90.1	87.7	72.6	79.4
Ours	×	**83.7**	**98.8**	**87.8**	**94.4**	**66.1**	64.7	72.3	78.8	**82.6**	94.2	73.9	96.1	**88.6**	**75.9**	**96.6**	80.2	**93.8**	**91.6**	**74.3**	**79.5**

Table 4. Per-category results on the Cityscapes test set. Note that G-SCNN and our method are trained with **only fine annotated data**. We achieve the state-of-the-art results with **83.7** mIoU. Best view on screen and zoom in.

More visualization improvement analysis: In Fig. 1, we include more visual comparisons on FCN and Deeplabv3+ with our methods. The right figures are our method's outputs. Our method solves the inner blur problem in large patterns for FCN and fixes missing details and inconsistent results on small objects on Deeplabv3+.

More visualization on decoupled feature representations and predictions: We give more visualization examples on decouple feature representation in Fig. 2.

More visualization on predicted flow fields: We also give more flow visualization examples in Fig. 3. The flow color encoding is shown below. The left part is the colormap, while the right part is the direction map.

More predicted fine-grained mask visualization: Fig. 4 gives more fine-grained mask prediction which are shown in the red boxes.

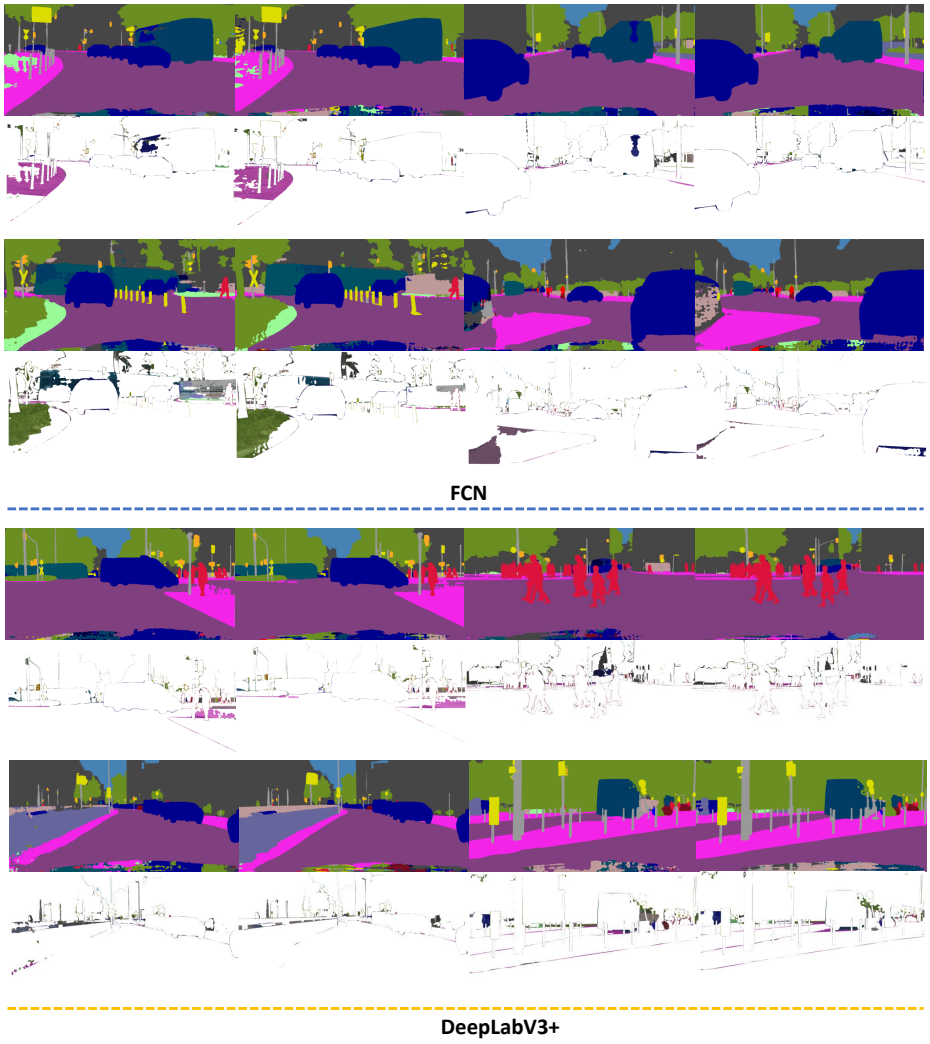

Fig. 1. Improvements over FCN (First four rows) and Deeplabv3+ (Last four rows). Best view it in color and zoom in.

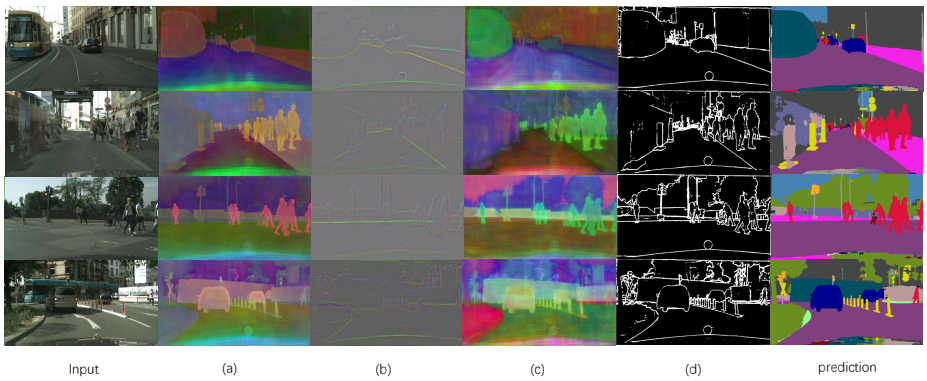

Fig. 2. More examples on Decoupled Feature Representation.(a) is F_{body}. (b) is $F - F_{body}$. (c) is re-constructed feature $\hat{F}$. (d) is edge prior prediction b with $t_b = 0.8$. Best view it in color and zoom in.

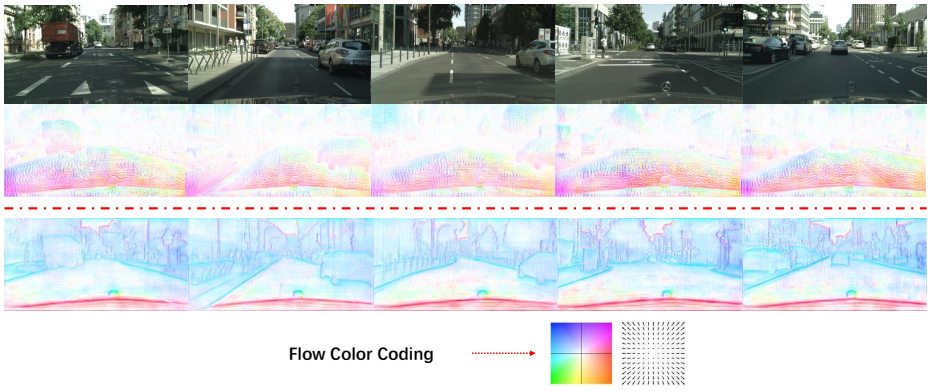

Fig. 3. Flow field visualizations. The first row shows the input images. The second row shows the generated flow fields based on FCN, while the third row shows the generated fields based on Deeplabv3+. We show flow directions and the color map in the last row.

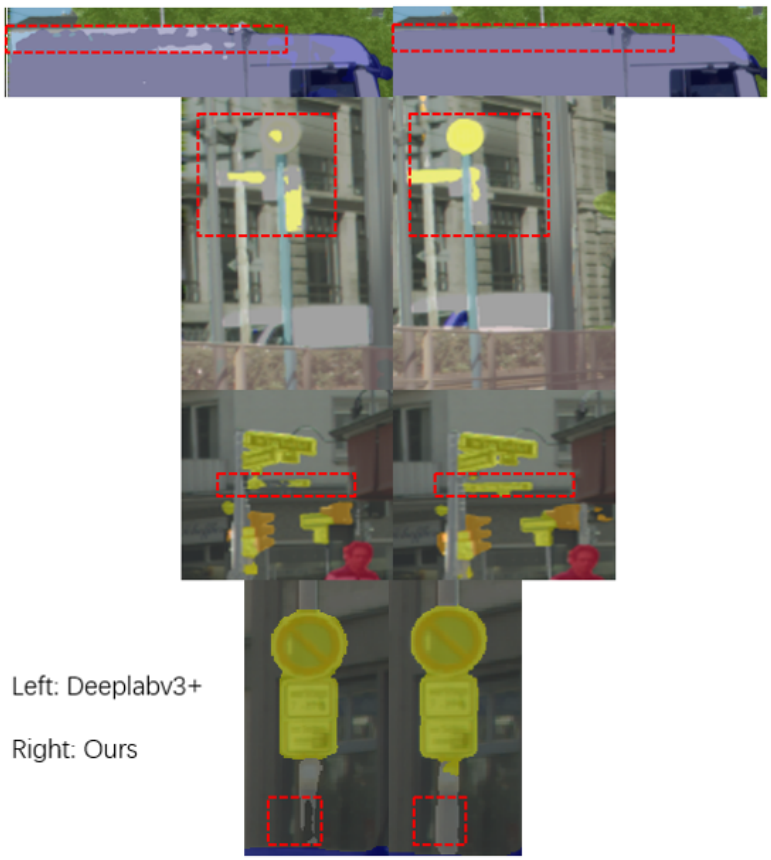

Fig. 4. Mask prediction examples based on Deeplabv3+. The refined parts are shown in red boxes.

References

1. Liu, S., De Mello, S., Gu, J., Zhong, G., Yang, M.H., Kautz, J.: Learning affinity via spatial propagation networks. In: NeurIPS. (2017)
2. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: ICCV. (2017)
3. Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., Loy, C.C., Lin, D.: mmdetection. https://github.com/open-mmlab/mmdetection (2018)
4. Takikawa, T., Acuna, D., Jampani, V., Fidler, S.: Gated-scnn: Gated shape cnns for semantic segmentation. ICCV (2019)
5. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: CVPR. (2016)
6. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR. (2017)
7. Ke, T.W., Hwang, J.J., Liu, Z., Yu, S.X.: Adaptive affinity fields for semantic segmentation. In: ECCV. (2018)
8. Yang, M., Yu, K., Zhang, C., Li, Z., Yang, K.: Denseaspp for semantic segmentation in street scenes. In: CVPR. (2018)
9. Fu, J., Liu, J., Tian, H., Fang, Z., Lu, H.: Dual attention network for scene segmentation. arXiv preprint (2018)
10. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint (2017)
11. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: ECCV. (2018)
12. Liu, C., Chen, L.C., Schroff, F., Adam, H., Hua, W., Yuille, A., Fei-Fei, L.: Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. CVPR (2019)
13. Chen, L.C., Collins, M., Zhu, Y., Papandreou, G., Zoph, B., Schroff, F., Adam, H., Shlens, J.: Searching for efficient multi-scale architectures for dense image prediction. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., eds.: NeurIPS. (2018)