Semantic Flow for Fast and Accurate Scene Parsing Supplemental Material

Xiangtai Li[1] *, Ansheng You[1] *, Zhen Zhu[2], Houlong Zhao[3], Maoke Yang[3], Kuiyuan Yang[3], Shaohua Tan[1], and Yunhai Tong[1]

[1] Key Laboratory of Machine Perception, MOE, School of EECS, Peking University
[2] Huazhong University of Science and Technology
[3] DeepMotion

Our supplemental material contains two parts. One is the more details on Cityscapes datasets and the other is the detailed setting on other datasets. We will opensource the our codebase.

1 Supplemental Experiments on Cityscapes

Detailed improvement on baseline models: Table 1 compares the detailed results of each category on the validation set, where ResNet-101 is used as backbone, and FPN decoder with PPM head serves as the baseline. Our method improves almost all categories, especially for 'truck' with more than 19% mIoU improvement.

More structured feature visualization on FAM: We give more structured feature visualization in Figure 1. We visualize more FAM outputs with two different location: last stages(below the blue line) and next to the last stage(above the blue line). For both cases, our module aligns the features into more structured representations with more clear shape and accurate boundaries.

More Visualization of Learned Flow: We also give more learned semantic flow Visualization in Figure 2.

More training details using Mapillary Vistas [5]: Mapillary Vistas is a large-scale dataset captured at street scenes, which contains 18K/2K/5K images for training, validation and testing, respectively. The dataset is similar to Cityscapes. Due to the larger variance of image resolutions than Cityscapes, we resize longer side to 2048 before data augmentation. To verify the performance improvement of SFNet by using more training data, we first pre-train SFNet on Mapillary Vistas for 50,000 iterations by using both train and val dataset, then finetune the model on Cityscapes for 50,000 iterations using Cityscapes fine-annotated data with the same setting in the paper before the submission to the test server.

Detailed setting about TensorRT The testing environment is TensorRT 6.0.1 with CUDA 10.1 on a single GTX 1080Ti GPU. In addition, we re-implement grid sampling operator by CUDA to be used together with TensorRT. The operator is provided by PyTorch and used in warping operation in the Flow Alignment

* The first two authors have equal contribution.

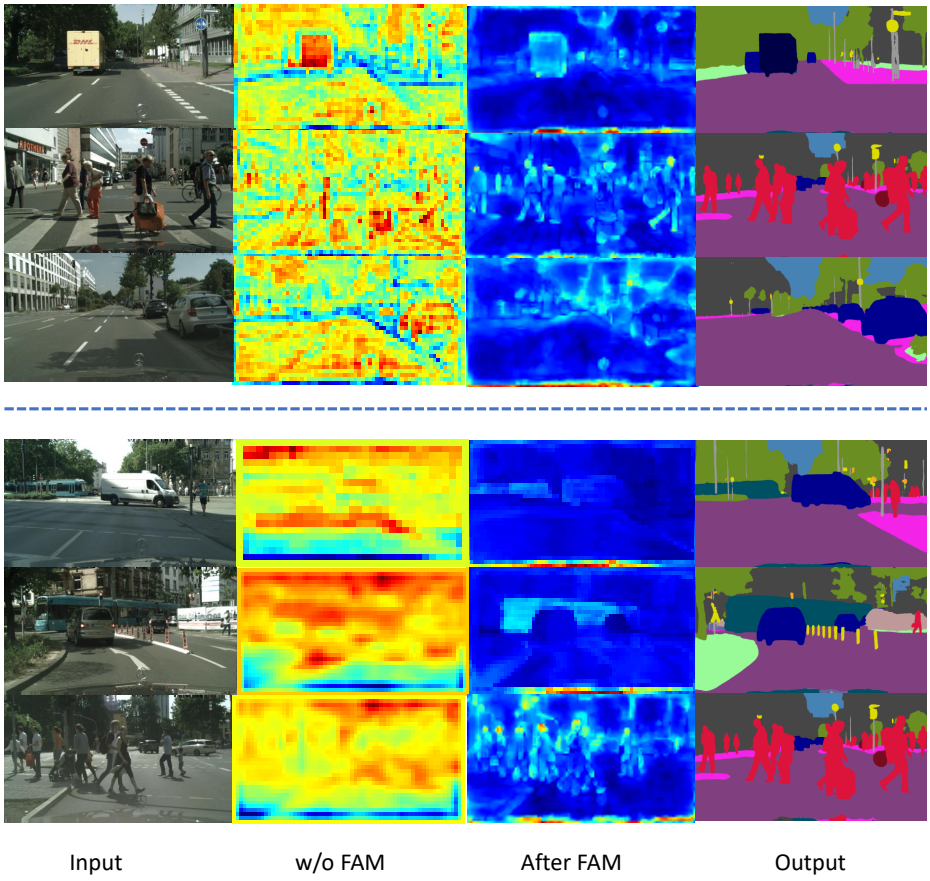

Fig. 1. More visualization of the aligned feature representation. The figures below the blue line are the outputs of last stage of FAM while the figures above the blues are the outputs of next to the last stage FAM with more fine details. Best view it on screen.

Module. Also, we also test the our speed on 1080-Ti using pytorch-library [6] where we report average time of inferencing 100 images.

Detailed results and settings on Cityscapes compared with accurate models: We give more detailed results in Table 2 for the state-of-the-art model comparison. For the fair comparison, we adopt multi-scale inference with 7 scales 0.5,0.75, 1, 1.25, 1.5, 1.75, 2.0 with flip operation. As shown in Table 2, our SFNet achi eves the best performance with less GFlops which has been calculated in the main paper.

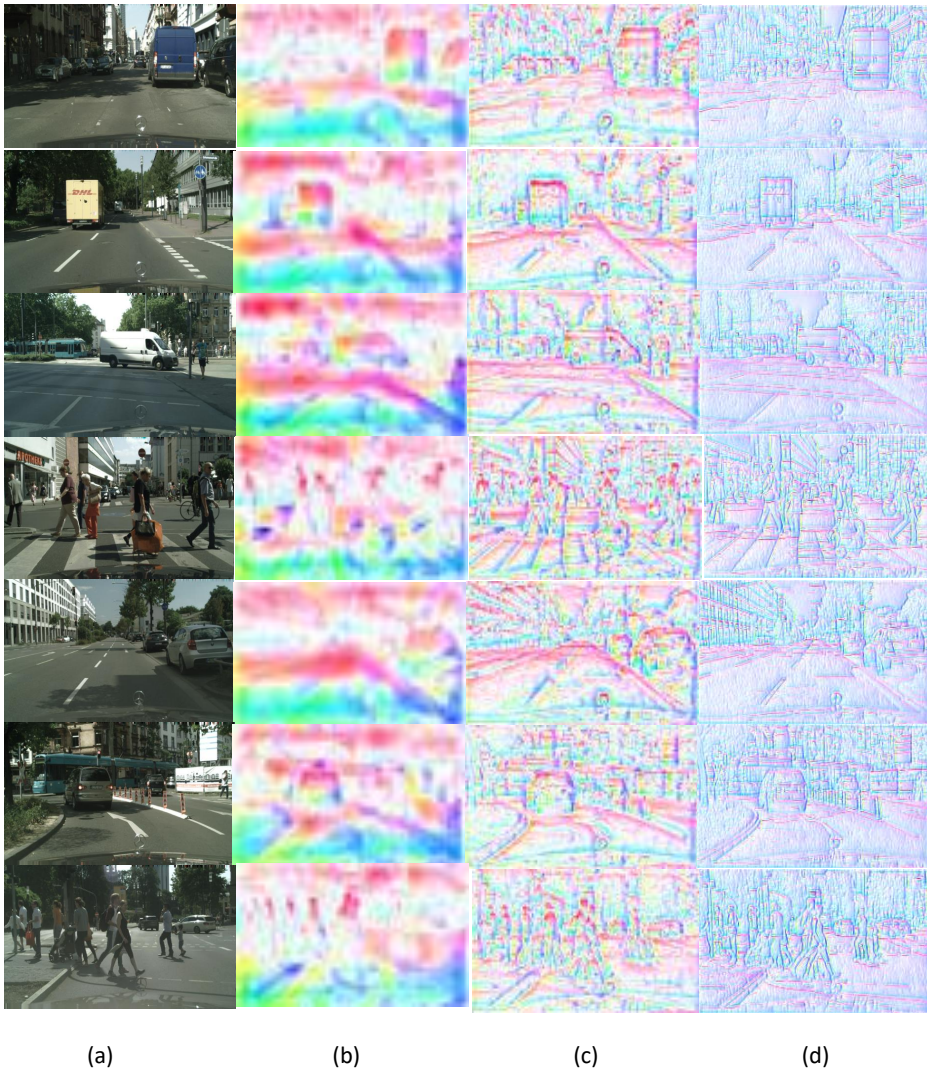

Fig. 2. More visualization of the learned semantic flow fields. Column (a) lists input images. Column (b)-(d) show the semantic flow of the three FAMs in an ascending order of resolution during the decoding process. Best view it on screen.

2 Detailed Experiment Settings on Other Datasets:

PASCAL Context: provides detailed semantic labels for whole scenes, and contains 4998 images for training and 5105 images for validation. We train the network for 120 epochs with batch size 16, crop size 512 with initial learning

Method	road	swalk	build	wall	fence	pole	tlight	sign	veg.	terrain	sky	person	rider	car	truck	bus	train	mbike	bike	mIoU
BaseLine	98.1	84.9	92.6	54.8	62.2	66.0	72.8	80.8	92.4	60.6	94.8	83.1	66.0	94.9	65.9	83.9	70.5	66.0	78.9	77.6
w/ FAM	98.3	85.9	93.2	62.2	67.2	67.3	73.2	81.1	92.8	60.5	95.6	83.2	65.0	95.7	84.1	89.6	75.1	67.7	78.8	79.8

Table 1. Quantitative per-category comparison results on Cityscapes validation set, where ResNet-101 backbone with the FPN decoder and PPM head serves as the strong baseline. Sliding window crop with horizontal flip is used for testing. Obviously, FAM boosts the performance of almost all the categories.

Method	road	swalk	build	wall	fence	pole	tlight	sign	veg.	terrain	sky	person	rider	car	truck	bus	train	mbike	bike	mIoU
ResNet38 [7]	98.5	85.7	93.0	55.5	59.1	67.1	74.8	78.7	93.7	72.6	95.5	86.6	69.2	95.7	64.5	78.8	74.1	69.0	76.7	78.4
PSPNet [11]	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	78.4
AAF [4]	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	79.1
SegModel [2]	98.6	86.4	92.8	52.4	59.7	59.6	72.5	78.3	93.3	72.8	95.5	85.4	70.1	95.6	75.4	84.1	75.1	68.7	75.0	78.5
DFN [10]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	79.3
BiSeNet [9]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	78.9
DenseASPP [8]	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	80.6
BFPNet [1]	98.7	87.1	93.5	59.8	63.4	68.9	76.8	80.9	93.7	72.8	95.5	87.0	72.1	96.0	77.6	89.0	86.9	69.2	77.6	81.4
DANet [3]	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	81.5
SFNet	98.8	87.1	93.6	63.2	62.7	68.4	75.6	80.3	93.8	71.0	95.7	87.7	73.2	96.5	75.9	92.3	89.5	71.4	78.0	**81.8**

Table 2. Per-category results on Cityscapes test set. Note that all the models are trained with only fine annotated data. Our method achieves **81.8%** mIoU with **much less** GFlops.

rate 1e-3. For evaluation, we perform multi-scale testing with horizontal flip operation.

ADE20k: is a more challenging scene parsing dataset annotated with 150 classes, and it contains 20K/2K images for training and validation. It has the various objects in the scene. We train the network for 120 epochs with batch size 16, crop size 512 and initial learning rate 1e-2. For final testing, we perform multi-scale testing with horizontal flip operation.

CamVid: is a road scene image segmentation dataset, which provides pixel-wise annotations for 11 semantic categories. There are 367 training images, 101 validation images and 233 testing images. We train the model with 120 epochs and our crop size is set to 640 and learning rate is 1e-3. The batch size is set to 16. For the final testing, we perform the single scale test for the fair comparison.

References

1. Ding, H., Jiang, X., Liu, A.Q., Magnenat-Thalmann, N., Wang, G.: Boundary-aware feature propagation for scene segmentation (2019)
2. Falong Shen, Gan Rui, S.Y., Zeng, G.: Semantic segmentation via structured patch prediction, context crf and guidance crf. In: CVPR (2017)
3. Fu, J., Liu, J., Tian, H., Fang, Z., Lu, H.: Dual attention network for scene segmentation. arXiv preprint arXiv:1809.02983 (2018)
4. Ke, T.W., Hwang, J.J., Liu, Z., Yu, S.X.: Adaptive affinity fields for semantic segmentation. In: ECCV (2018)
5. Neuhold, G., Ollmann, T., Rota Bulo, S., Kontschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: ICCV (2017)
6. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch. In: NIPS-W (2017)
7. Wu, Z., Shen, C., van den Hengel, A.: Wider or deeper: Revisiting the resnet model for visual recognition. arXiv preprint arXiv:1611.10080 (2016)
8. Yang, M., Yu, K., Zhang, C., Li, Z., Yang, K.: Denseaspp for semantic segmentation in street scenes. In: CVPR (2018)
9. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Bisenet: Bilateral segmentation network for real-time semantic segmentation. In: ECCV (2018)
10. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Learning a discriminative feature network for semantic segmentation. In: CVPR (2018)
11. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR (2017)