Cars Can't Fly up in the Sky: Improving Urban-Scene Segmentation via Height-driven Attention Networks

A. Supplementary Material

This section complements our paper by presenting additional information, experimental results and visualizations. First, we provide further comparison results with other state-of-the-arts in Section A.1. In Section A.2, we conduct experiments to find the best way to incorporate positional information. We then describe the architecture details of the baseline and HANet in Section A.3. In Section A.4, we compare height-wise and width-wise class distributions. Finally, we conduct quantitative and qualitative comparisons between ours and the baseline model in Section A.5.

A.1. Additional comparisons with other models

Comparisons on Cityscapes validation set We compare the best performance of our model with other state-of-the-art models on the Cityscapes validation set.

Models (Year)	Backbone	mIoU(%)
ANLNet [8] ('19)	ResNet-101	79.9
DANet [1] ('19)	ResNet-101	81.5
CCNet [3] ('19)	ResNet-101	81.3
ACFNet [6] ('19)	ResNet-101	81.46
Ours	ResNet-101	**82.05**

Table 1: Comparisons against the best performances reported in the published papers of other state-of-the-art models on the Cityscapes validation set. The models based on ResNet-101 are compared.

A.2. Positional encoding and embedding.

In the NLP domains, there exist different approaches to inject positional information of each token in the input sequence. Positional encoding using sinusoidal values [5] and learned positional embeddings [2] have been shown to produce comparable performances [5]. We conduct experiments to find the best way to incorporate positional information. It turns out that the best way is to put sinusoidal positional encoding into the second convolutional layer of HANet (Table 2).

Methods	Injected layer	
	1st	2nd
Sinusoidal encoding	79.61%	**80.29%**
Learnable embedding (from scratch)	79.95%	79.60%
Learned embedding (from pretrained)	79.61%	79.30%

Table 2: Performances comparison with respect to the layers and methods of positional encoding. Note that HANet consists of three convolutional layers. ResNet-101, output stride 8 on Cityscapes validation set.

A.3. Further implementation details

We implement our methods based on the open-source implementations of NVIDIA semantic segmentation model [7]. HANet consists of three convolutional layers incorporating dropout and batch normalization. To extract the height-wise contextual information from each row, we empirically adopt average pooling.

HANet architecture Fig. 1 shows detailed architecture of HANet. Width-wise pooling and interpolation for coarse attention are implemented using two-dimensional adaptive average pooling operation[1] in Pytorch. Afterwards, a

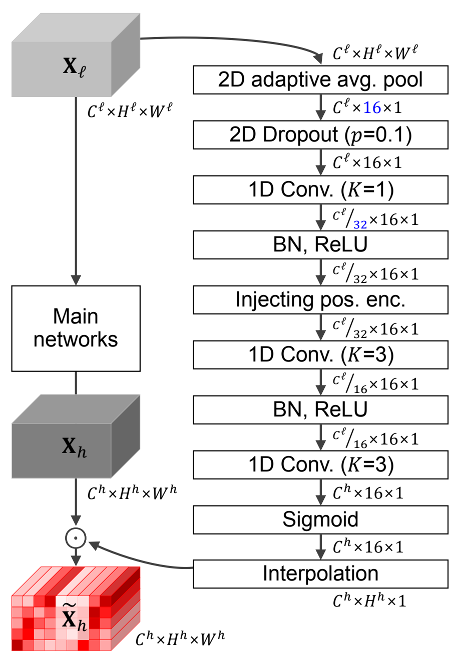

Figure 1: Detailed architecture of HANet. p denotes the dropout probability, and K denotes the kernel size of each one-dimensional convolution layer. BN denotes a batch normalization layer.

[1] `https://pytorch.org/docs/stable/nn.html#torch.nn.AdaptiveMaxPool2d`

Model	mIoU	road	swalk	build.	wall	fence	pole	tligh.	tsign	veg	terr.	sky	pers.	rider	car	truck	bus	train	mcyc	bcyc
Baseline	81.14	98.5	87.3	93.6	66.1	64.4	68.7	74.0	82.0	93.2	65.6	95.2	84.3	66.0	95.7	80.6	92.8	85.0	68.9	80.0
+HANet	82.05	98.6	87.7	93.7	66.7	**66.2**	68.7	74.4	81.9	93.3	**67.7**	95.3	84.5	66.9	96.1	**87.9**	92.7	**86.0**	**70.7**	80.1

Table 3: Performance comparison of our methods against the baseline in terms of per-class IoU and mIoU measures. Inference techniques such as sliding, multi-scale, and flipping are applied. ResNet-101, output stride 8 on the Cityscapes validation set.

dropout layer and three one-dimensional convolutional layers are applied. Blue values in Fig. 1, 16 and 32, are respectively the height of coarse attention and the channel reduction ratio r, which are our hyper-parameters. All the hyperparameters can be found in our code.

Baseline architecture Fig. 2 shows detailed architecture of the baseline model, which is based on DeepLabv3+. As an encoder-decoder architecture, low-level features obtained from ResNet stage 1 are concatenated to high-level features via skip-connection. An auxiliary loss proposed in PSPNet [4] is applied to facilitate the learning process of deep networks. To adopt the auxiliary loss, additional convolutional layers are added after ResNet stage 3 as an auxiliary branch. The loss for this auxiliary branch has a weight of 0.4. The output stride is set to 8 as shown in blue color; this can be set differently, e.g., 16.

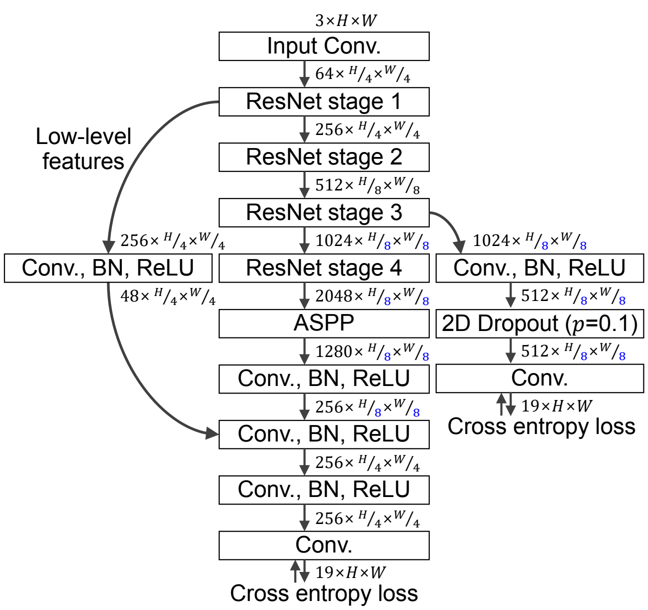

Figure 2: Detailed architecture of the baseline model

A.4. Height- and width-wise class distribution

As shown in Fig. 3, the width-wise class distributions are relatively similar across columns than the height-wise ones are, so it would be relatively difficult to extract distinct information with respect to the horizontal position of an image. Also, empirically, no meaningful performance increase has been observed when using the attention networks exploiting a width-wise class distribution.

This clear pattern corroborates the rationale behind the idea of HANet that extracts and incorporates height-wise contextual information rather than the width-wise one.

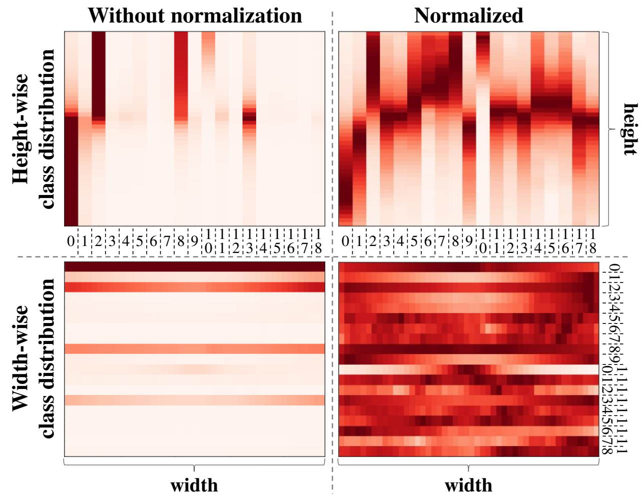

Figure 3: Comparison of a height-wise and a width-wise class distributions. A darker color indicates a higher probability (more pixels) assigned to a particular class (from 0 to 18). The height-wise class distributions show distinct patterns across vertical positions while it is not the case for width-wise ones. Normalized distributions of each class are presented on the right column.

A.5. Per-class IoU and segmentation maps

We present per-class IoU and segmentation maps to analyze HANet qualitatively and quantitatively.

Comparison to baseline. Table 3 shows the per-class IoU and mIoU results to compare the baseline and our methods in detail. Compared to the baseline, all the classes show similar or improved results; up to 7.3% IoU increase is observed. Qualitatively, ours can properly distinguish individual objects, even between the classes much alike to each other (Figs. 4 and 5). From our result in Fig. 4(a) and Fig. 5(a)-(c), one can see that the train, trucks, or cars in a far, crowded region are properly predicted by ours, even if similar vehicles are found nearby. Also, vegetation is accurately distinguished from terrain in ours, compared to the baseline (Fig. 4(b)). Another interesting examples are found in Fig. 5(e)-(f); the poles are connected fully in ours but dotted or missed in the baseline. We conjecture that HANet helps to distinguish confusing classes by properly gating the activation maps using height-wise contextual information based on their vertical positions. To summarize, compared to the baseline, our method generally forms a clear boundary of a object while avoiding its unnecessary fragmentation into multiple pieces.

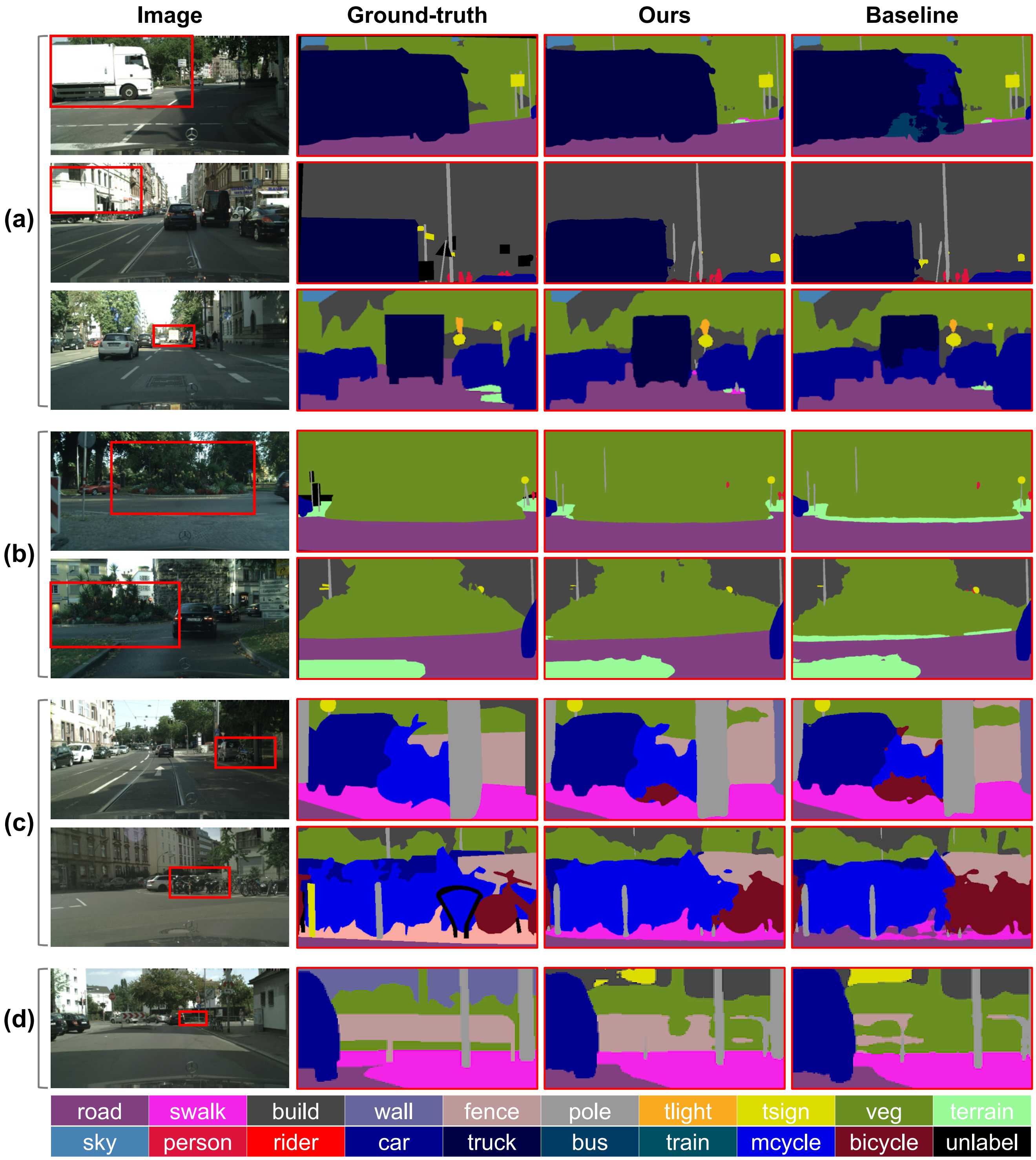

Figure 4: Comparison of predicted segmentation maps: (a) truck, bus, and car. (b) vegetation and terrain. (c) motorcycle and bicycle. (d) fence and vegetation.

Figure 5: Comparison of predicted segmentation maps: (a) truck and car. (b) car and truck. (c) bus and car. (d) building and fence. (e) sky and building; pole and building. (f) pole and building.

References

[1] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3146–3154, 2019.

[2] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In *Proc. the International Conference on Machine Learning (ICML)*, pages 1243–1252. JMLR. org, 2017.

[3] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proc. of the International Conference on Computer Vision (ICCV)*, pages 603–612, 2019.

[4] Hanchao Li, Pengfei Xiong, Jie An, and Lingxue Wang. Pyramid attention network for semantic segmentation. In *British Machine Vision Conference (BMVC)*, 2018.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017.

[6] Fan Zhang, Yanqin Chen, Zhihang Li, Zhibin Hong, Jingtuo Liu, Feifei Ma, Junyu Han, and Errui Ding. Acfnet: Attentional class feature network for semantic segmentation. In *Proc. of the International Conference on Computer Vision (ICCV)*, 2019.

[7] Yi Zhu, Karan Sapra, Fitsum A Reda, Kevin J Shih, Shawn Newsam, Andrew Tao, and Bryan Catanzaro. Improving semantic segmentation via video propagation and label relaxation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8856–8865, 2019.

[8] Zhen Zhu, Mengde Xu, Song Bai, Tengteng Huang, and Xiang Bai. Asymmetric non-local neural networks for semantic segmentation. In *Proc. of the International Conference on Computer Vision (ICCV)*, pages 593–602, 2019.