MobileStereoNet: Towards Lightweight Deep Networks for Stereo Matching (Appendix)

https://github.com/cogsys-tuebingen/mobilestereonet

This appendix provides more details of our work. Here, we have the following sections: 1: Detailed baseline architectures, 2: More qualitative results, 3: Incorporating light blocks in other modules, 4: Implementation details, and 5: Analyzing the complexity.

1. Detailed baseline architectures

The detailed architectures of the 2D and 3D baseline models are displayed in Fig. 1. The numbers in the blocks indicate the output size of each particular layer/module. The feature extraction step is the same for the two models. The architecture of hourglass and its intraconnections are also similar, except that in the 2D baseline, the convolutions are all in 2D type, while there are 3D convolutions in hourglass of the 3D baseline. These two models differ in the cost volume construction and the channel reduction module as well.

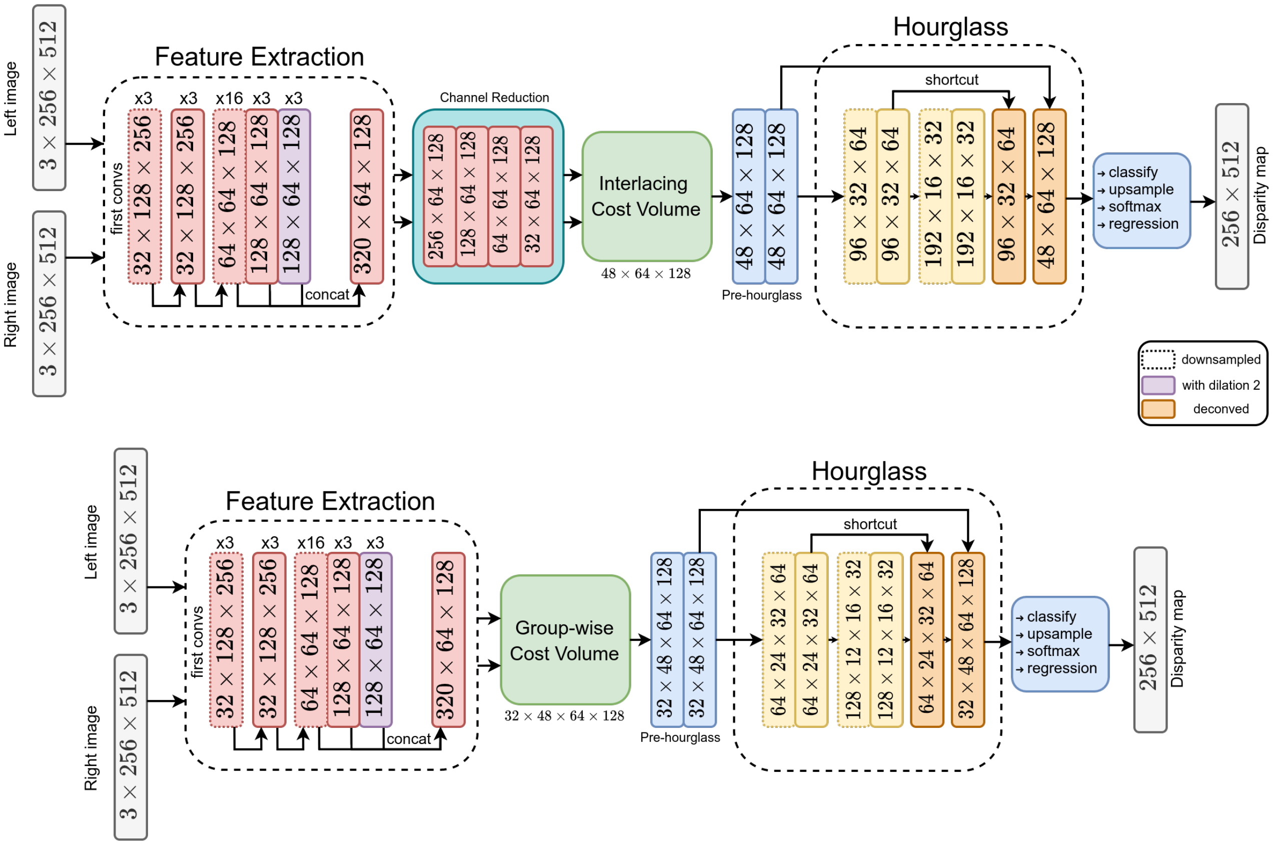


Figure 1: *Top*: 2D baseline, *Bottom*: 3D baseline. The numbers in the blocks indicate the output size of each particular layer/module.

2. More qualitative results

Figure 2 depicts more qualitative results on SceneFlow dataset. We have also shown qualitative comparison on KITTI 2015 validation set in Fig. 3.

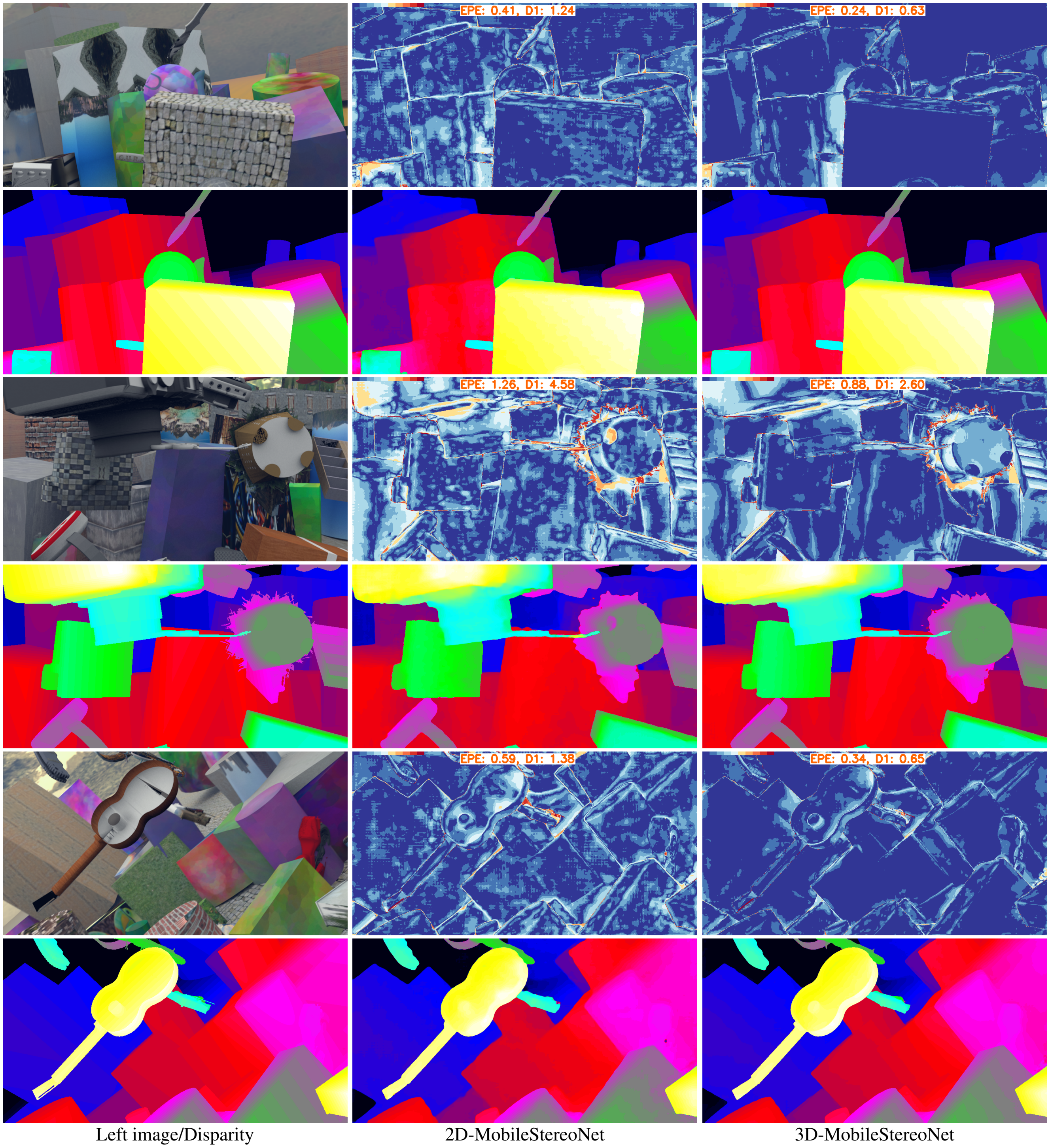
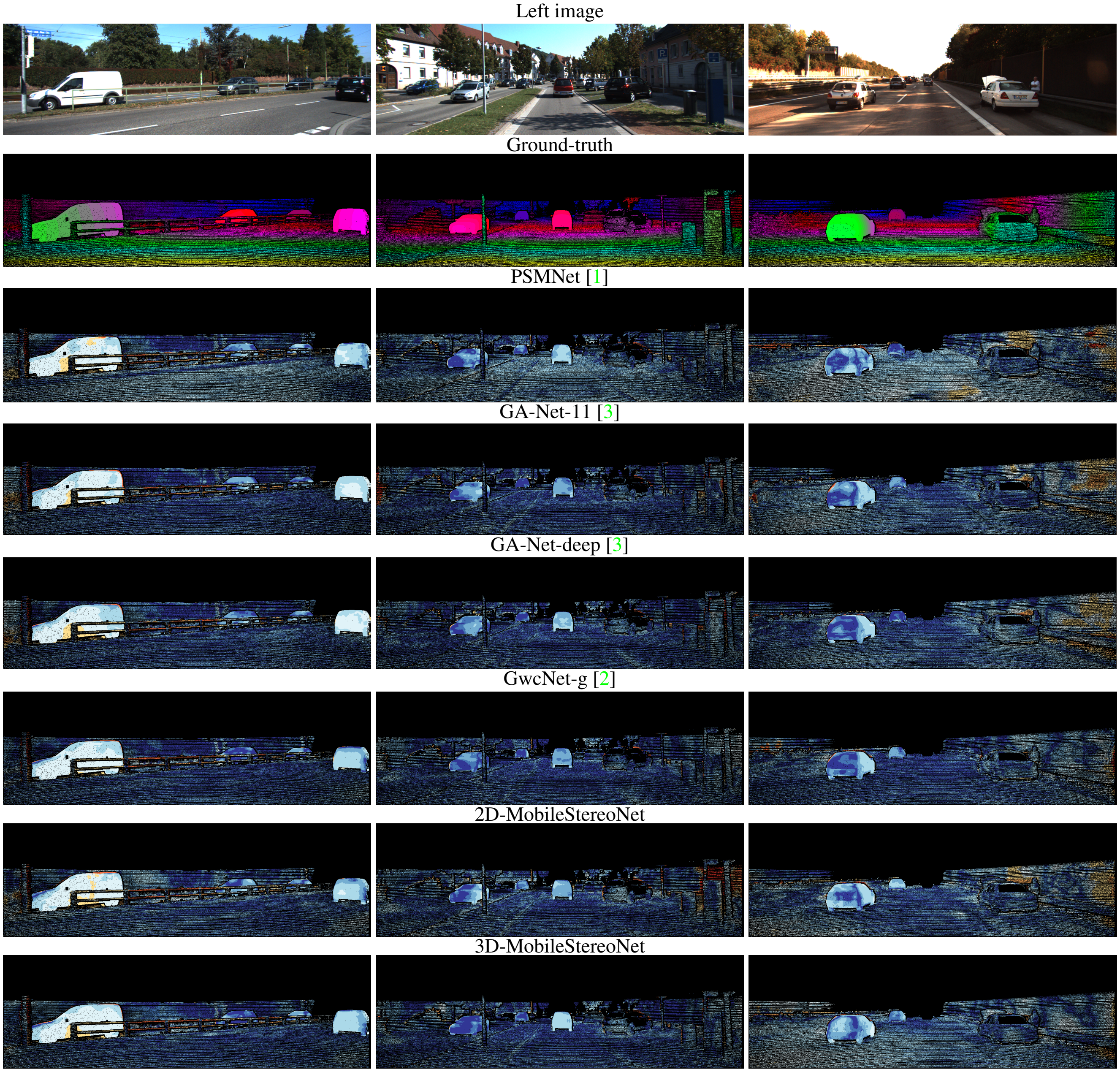


Figure 2: Qualitative performance on SceneFlow: Every two rows correspond to a test sample. In the left-most column, the samples and the ground-truth disparity maps are illustrated. The following two columns show the disparity and error maps (embedded with error values) estimated by 2D-MobileStereoNet and 3D-MobileStereoNet. Warmer colors in error maps denote higher errors.



Left image

Ground-truth

PSMNet [1]

GA-Net-11 [3]

GA-Net-deep [3]

GwcNet-g [2]

2D-MobileStereoNet

3D-MobileStereoNet

Figure 3: Qualitative performance on KITTI 2015 validation set: From top to bottom, the left image, the ground-truth disparity map and the estimated disparity maps by PSMNet [1], GA-Net-11 [3], GA-Net-deep [3], GwcNet-g [2], 2D-MobileStereoNet and 3D-MobileStereoNet are illustrated. For a fair comparison, we trained all the models with a 159/40 split of KITTI 2015 training test. Warmer colors in error maps denote higher errors.

From Fig. 3, once again, we can verify that 2D-MobileStereoNet shows close performance to 3D models with the least number of operations. Also, 3D-MobileStereoNet obtains competitive or better accuracy with the least number of parameters among other methods.

3. Incorporating light blocks in other modules

As mentioned in the paper, in order to further reduce the complexity, the first convolutions in the feature extraction and the pre-hourglass convolutions (*cf.* Fig. 1) are replaced with MobileNet-V2 (v_2). The experimental results are reported in

Tables 1 and 2. Note that the first convolutions are of the 2D type for both 2D and 3D baselines; however, the pre-hourglass comes in 2D or 3D convolutions depending on the baseline. We can observe that in 2D-MobileStereoNet, when the two modules are replaced with MobileNet-V2 (v_2), the network obtains the least EPE. In 3D-MobileStereoNet, this combination yields slightly higher EPE. However, due to the nice reduction in the computation cost, we consider the same design choice for the 3D network. It is noteworthy that we have examined MobileNet-V1 (v_1) for these modules as well. However, as it deteriorates the performance, we ignore v_1 for these modules, albeit it shows much decrease in the cost.

first-conv$_{2D}$	pre-HG$_{2D}$	EPE(px) ↓	MACs(G) ↓	Params(M) ↓
conv.	conv.	1.50	30.33	1.21
conv.	v_2	1.41	30.0	1.16
v_2	conv.	1.54	29.75	1.20
v_2	v_2	1.40	29.42	1.15

Table 1: Performance evaluation for the selected variant of 2D baseline (FE$_{2D}$:v_1, HG$_{2D}$:v_2) from Tab. 3a of the paper, when replacing other components with v_2 block ($t = 2$).

first-conv$_{2D}$	pre-HG$_{3D}$	EPE(px) ↓	MACs(G) ↓	Params(M) ↓
conv.	conv.	0.99	105.01	0.98
conv.	v_2	1.01	69.44	0.89
v_2	conv.	0.99	104.44	0.97
v_2	v_2	1.01	68.86	0.88

Table 2: Performance evaluation for the selected variant of 3D baseline (FE$_{2D}$:v_1, HG$_{3D}$:v_2) from Tab. 3b of the paper, when replacing other components with v_2 block ($t = 2$).

4. Implementation details

We used PyTorch for implementation and conducting experiments. All the trainings are executed on 4 × NVIDIA GeForce GTX 1080 Ti. We adapt the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. On the SceneFlow dataset, the networks are trained for 20 epochs, starting with a learning rate of 0.001. The learning rate is halved after epoch 10, 12, 14, and 16. The best model is selected based on the least EPE value. In the experiments on the KITTI 2015 validation set, we finetune the best SceneFlow model for 400 epochs, reducing the initial learning rate 0.001 by a factor of 10 after 200 epochs. To submit the results to the KITTI 2015 benchmark, we finetune starting from a SceneFlow checkpoint showing the best generalization performance from the SceneFlow to the KITTI 2015 images. For the 3D-MobileStereoNet, we used a batch size of 4, and for 2D-MobileStereoNet, the batch size is 8.

5. Analyzing the complexity

Table 3 shows the computation cost of the main modules, *i.e.* feature extraction and encoder-decoder, in baselines (with standard convolutions) and in MobileStereoNets. Note that feature extraction is the same in 2D and 3D models. We see our design choice for feature extraction is significantly reducing the complexity both in operation (from 52.07 to 7.84 GigaMACs) and in parameters (from 7.84 to only 0.39 million). We also observe that the cost of the encoder-decoder modules, either in 2D or 3D, is reduced in lighter networks in both number of operations and parameters. Evidently, the major bottleneck for the 3D models is the encoder-decoder with 3D convolutions.

	Baselines		MobileStereoNets	
	MACs(G)	Params(M)	MACs(G)	Params(M)
Feature Extraction	52.07	2.95	7.84	0.39
Encoder-decoder$_{2D}$ in 2D-MobileStereoNet	4.38	2.61	3.92	1.64
Encoder-decoder$_{3D}$ in 3D-MobileStereoNet	167.51	3.45	128.73	1.34

Table 3: Analyzing the computation cost in terms of MACs and number of parameters for the main modules.

References

[1] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018.

[2] Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li. Group-wise correlation stereo network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3273–3282, 2019.

[3] Feihu Zhang, Victor Prisacariu, Ruigang Yang, and Philip HS Torr. GA-Net: Guided aggregation net for end-to-end stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 185–194, 2019.