Three Ways to Improve Semantic Segmentation with Self-Supervised Depth Estimation
(Supplementary Materials)

Lukas Hoyer
ETH Zurich
lhoyer@student.ethz.ch

Dengxin Dai
ETH Zurich
dai@vision.ee.ethz.ch

Yuhua Chen
ETH Zurich
yuhua.chen@vision.ee.ethz.ch

Adrian Köring
University of Bonn
adrian.koering@uni-bonn.de

Suman Saha
ETH Zurich
suman.saha@vision.ee.ethz.ch

Luc Van Gool
ETH Zurich & KU Leuven
vangool@vision.ee.ethz.ch

A. Further Implementation Details

In the following paragraphs, a more detailed description of the network architecture and the training is provided. The reference implementation is available at https://github.com/lhoyer/improving_segmentation_with_selfsupervised_depth.

Network Architecture The neural network combines a DeepLabv3 [3] with a U-Net [10] decoder for depth and segmentation prediction each. As encoder, a ResNet101 with dilated (instead of strided) convolutions in the last block is used, following [3]. Features from multiple scales are aggregated by an ASPP [3] block with dilation rates of 6, 12, and 18. Similar to U-Net [10], the decoder has five upsampling blocks with skip connections. Each upsampling block consists of a 3x3 convolution layer (except the first block, which is the ASPP), a bilinear upsampling operation, a concatenation with the encoder features of the corresponding size (skip connection), and another 3x3 convolution layer. Both convolutional layers are followed by an ELU non-linearity. The number of output channels for the blocks are 256, 256, 128, 128, and 64. The last four blocks also have another 3x3 convolutional layer followed by a sigmoid activation attached to their output for the purpose of predicting the disparity at the respective scale. For effective multi-task learning, we additionally follow PAD-Net [11] and deploy an attention-guided multi-modal distillation module with additional side output for semantic segmentation after the third decoder block. In experiments without multi-task learning, only the semantic segmentation decoder is used. For pose estimation, we use a lightweight ResNet18 encoder followed by four convolutions to produce the translation and the rotation in angle-axis representation as suggested in [5].

Table S1. Training and inference time on an Nvidia Tesla P100 averaged over 100 iterations or 500 images, respectively. D-T: SDE Transfer Learning, D-M SDE Transfer and Multi-Task Learning, P: Pseudo-Labelling, X-D: Mix Depth

D	P	X	Training Time	Inference Time
T			188 ms/it	66 ms/img
T	✓		466 ms/it	67 ms/img
T	✓	D	476 ms/it	66 ms/img
M	✓	D	1215 ms/it	160 ms/img

Runtime To give an impression of the computational complexity of our architecture, we provide the training time per iteration and the inference time per image on an Nvidia Tesla P100 in Tab. S1. The values are averaged over 100 iterations or 500 images, respectively. Please note that these timings include the computational overhead of the training framework such as logging and validation metric calculation.

Data Selection In the data selection experiment, we use a slimmed network architecture for f_{SIDE} with a ResNet50 backbone, 256, 128, 128, 64, and 64 decoder channels, and BatchNorm [6] in the decoder for efficiency and faster convergence. The depth student network is trained using a berHu loss [12, 8]. The quality of the selected subset with annotations $\mathcal{G}_A$ is evaluated for semantic segmentation using our default architecture and training hyperparameters.

B. Cross-Dataset Transfer Learning

In this section, we show that the unlabeled image sequences and the labeled segmentations can also originate from different datasets within similar visual domains. For that purpose, we train the SDE on Cityscapes sequences

Table S2. Performance on the CamVid test set (mIoU in %, standard deviation over 3 random seeds). The SDE is trained on Cityscapes sequences. DT: SDE Transfer Learning, XD - DepthMix, S: Data Selection.

# Labeled	50	↴	100	↴	367 (Full)	↴
Baseline	59.16 ±1.79	↴	63.05 ±0.59	↴	68.18 ±0.13	↴
Ours (DT)	62.75 ±2.32	+3.60	66.19 ±0.96	+3.15	70.45 ±0.35	+2.27
ClassMix [9]	65.89 ±0.33	+6.73	67.48 ±1.02	+4.43	-	
Ours (DT+XD)	66.82 ±1.16	+7.66	68.91 ±0.62	+5.86	71.46 ±0.22	+3.29
Ours (DT+XD+S)	68.23 ±0.39	+9.07	69.62 ±0.64	+6.57	-	

and learn the semi-supervised semantic segmentation on the CamVid dataset [1], which contains 367 train, 101 validation, and 233 test images with dense semantic segmentation labels for 11 classes from street scenes in Cambridge. To ensure a similar feature resolution, we upsample the CamVid images from 480×360 to 672×512 pixels and randomly crop to a size of 512×512.

Table S2 shows that the results on CamVid are similar to our main results on Cityscapes. For 50 labeled training samples, SDE pretraining improves the mIoU by 3.6 percentage points, pseudo-labels and DepthMix by another 4.07 percentage points, and data selection by another 1.41 percentage points. In the end, our proposed method significantly outperforms ClassMix by 2.34 percentage points for 50 labeled samples and 2.14 percentage points for 100 labeled samples. Also for the fully labeled dataset, our method can improve the performance by 3.29 percentage points.

C. Further Example Predictions

Further examples for semantic segmentation and SDE are shown in Fig. S1. In general, the same observations as in the main paper can be made. Our method provides clearer segmentation contours for objects that are bordered by pronounced depth discontinuities such as pole, traffic sign, or traffic light. We also show improved differentiation between similar classes such as truck, bus, and train. On the downside, SDE sometimes fails for cars driving directly in front of the camera (see 7th row in Fig. S1) and violating the reconstruction assumptions. Those cars are observed at the exact same location across the image sequence and can not be correctly reconstructed during SDE training, even with correct depth and pose estimates. However, this differentiation between moving and non-moving cars does not hinder the transfer of SDE-learned features to semantic segmentation but can cause problems with DepthMix (see Section D).

D. DepthMix Real-World Examples

In Fig. S2, we show examples of DepthMix applied to Cityscapes crops. Generally, it can be seen that DepthMix works well in most cases. The self-supervised depth estimates allow to correctly model occlusions and the produced synthetic samples have a realistic appearance.

In Fig. S3, we show a selection of typical failure cases of DepthMix. First, the SDE can be inaccurate for dynamic objects (see Sec. C), which can cause an inaccurate structure within the mixed image (Fig. S3 a, b, and c). However, this type of failure case is common in ClassMix and its frequency is greatly reduced with DepthMix. A remedy might be SDE extensions that incorporate the motion of dynamic objects [2, 4, 7]. Second, in some cases, the SDE can be imprecise and the depth discontinuities do not appear at the same location as the class border. This can cause artifacts in the mixed image (Fig. S3 d and e) but also in the mixed segmentation (Fig. S3 e: sky within the building). Note that the same can happen for ClassMix when using pseudo-labels for creating the mix mask.

References

[1] Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, pages 88–97, 2009.

[2] Vincent Casser, Soeren Pirk, Reza Mahjourian, and Anelia Angelova. Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. In *AAAI Conf. Artif. Intell.*, pages 8001–8008, 2019.

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 834–848, 2017.

[4] Qi Dai, Vaishakh Patil, Simon Hecker, Dengxin Dai, Luc Van Gool, and Konrad Schindler. Self-supervised object motion and depth estimation from video. In *IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, pages 1004–1005, 2020.

[5] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Int. Conf. Comput. Vis.*, pages 3828–3838, 2019.

[6] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[7] Marvin Klingner, Jan-Aike Termöhlen, Jonas Mikolajczyk, and Tim Fingscheidt. Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance. In *Eur. Conf. Comput. Vis.*, pages 582–600, 2020.

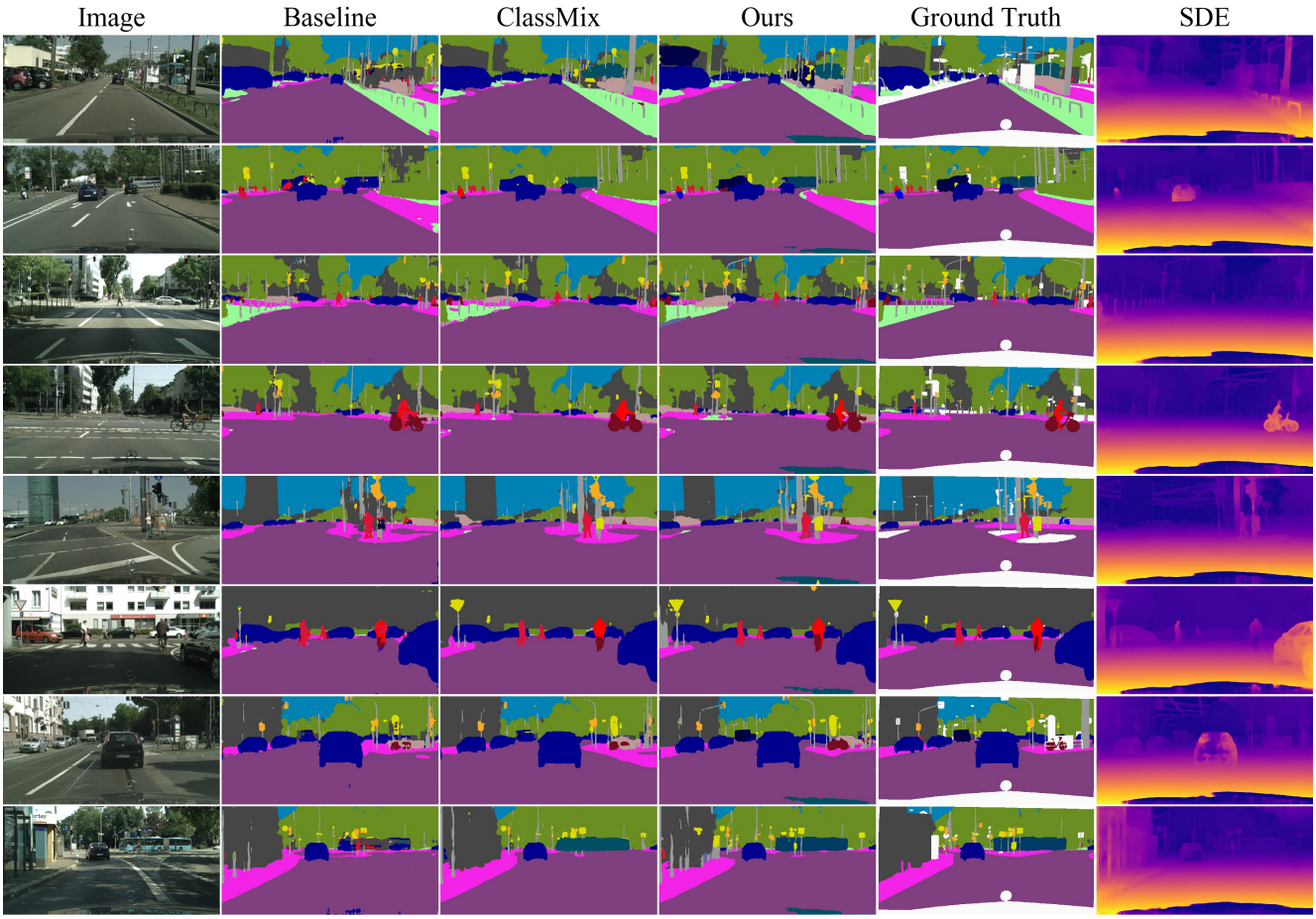

Figure S1. Further example predictions for 100 annotated training samples including the self-supervised disparity estimate of the multi-task learning framework.

[8] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *Int. Conf. 3D Vision*, pages 239–248, 2016.

[9] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *IEEE Winter Conf. on Applications of Comput. Vis.*, pages 1369–1378, 2021.

[10] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Int. Conf. Medical Image Computing and Computer-assisted Intervention*, pages 234–241, 2015.

[11] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 675–684, 2018.

[12] Laurent Zwald and Sophie Lambert-Lacroix. The berhu penalty and the grouped effect. *arXiv preprint arXiv:1207.6868*, 2012.

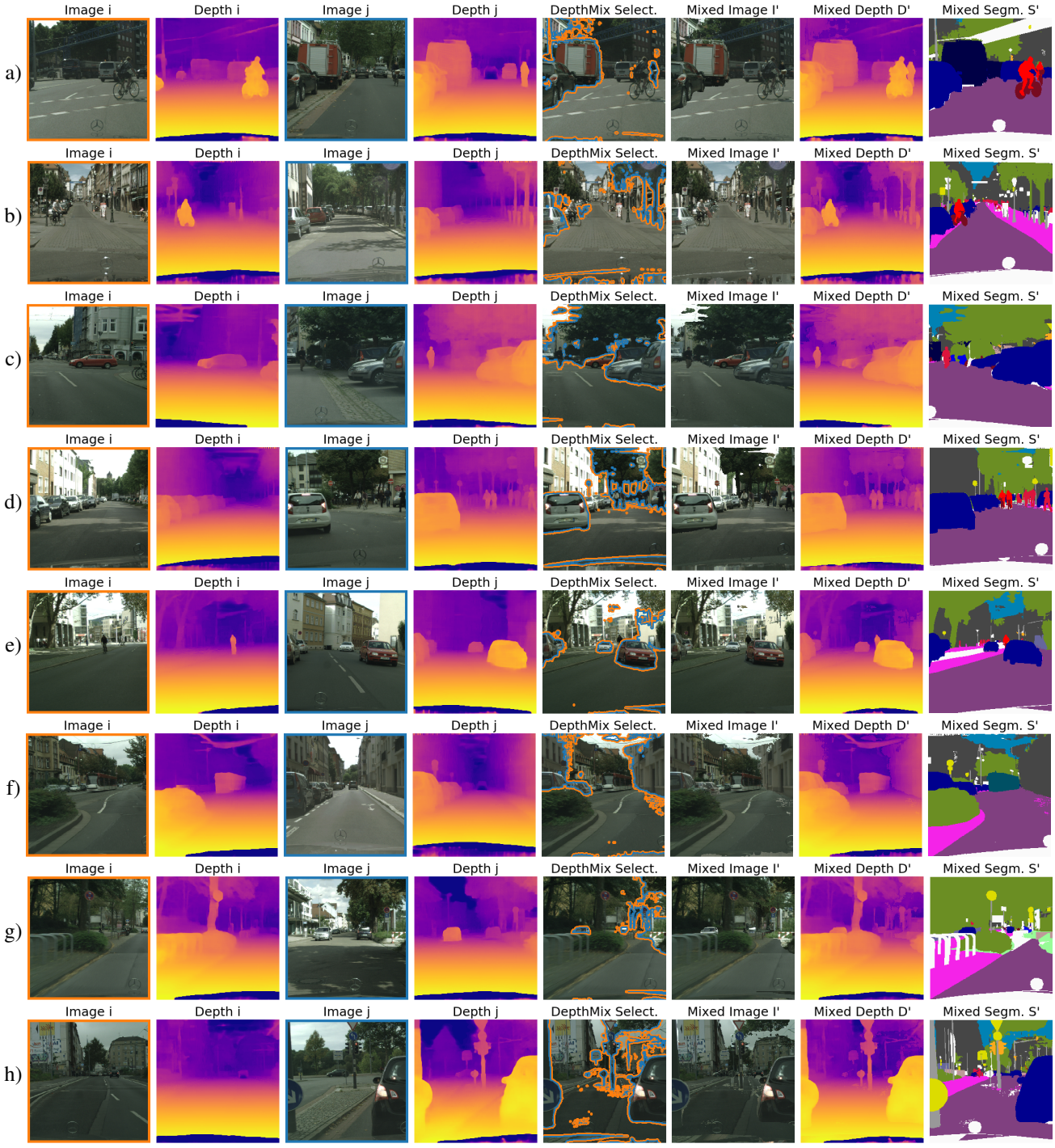

Figure S2. DepthMix applied to Cityscapes crops. From left to right, the source images with their SDE estimate, the mixed image I' overlaid with border of the mix mask M in blue/orange depending on the adjacent source image (i - orange, j - blue), the mixed image without visual guidance I', the mixed depth D', and the mixed segmentation S' are shown. For simplicity, the source segmentations for the mixed segmentation S' originate from the ground truth labels.

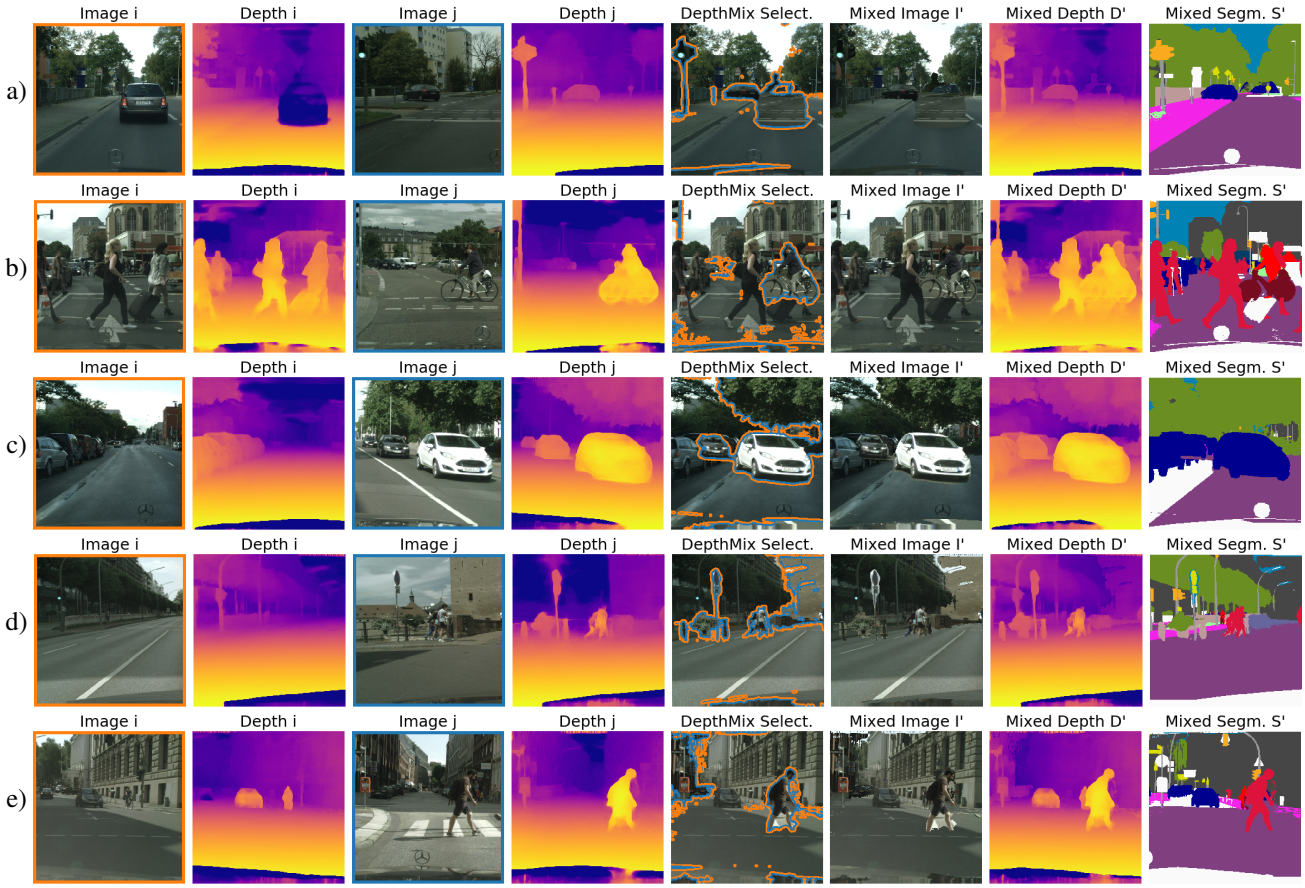

Figure S3. DepthMix failure cases. From left to right, the source images with their SDE estimate, the mixed image I' overlaid with border of the mix mask M in blue/orange depending on the adjacent source image (i - orange, j - blue), the mixed image without visual guidance I', the mixed depth D', and the mixed segmentation S' are shown. For simplicity, the source segmentations for the mixed segmentation S' originate from the ground truth labels.