Supplementary Material for *Do What You Can, With What You Have: Scale-aware and High Quality Monocular Depth Estimation Without Real World Labels*

Kunal Swami, Amrit Muduli, Uttam Gurram, Pankaj Bajpai
Camera Solutions Group,
Samsung Research India Bangalore
{kunal.swami,amrit.muduli,uttam.g,pankaj.b}@samsung.com

Contents

1. Network Details

1.1. Depth and Pose Network

As the main focus of this work is to develop a training methodology, we adopt the light-weight network architecture from Monodepth2 [6] as our MDE and pose estimation network. More specifically, we use ResNet18 [8] as our depth as well as pose encoder, whereas a DispNet [11] based decoder is used for the MDE model. The architecture of our depth and pose decoders are explained in Tab. 1 and Tab. 2 respectively, these tables are taken directly from Monodepth2 [6] and added here to make the material self-contained.

1.2. ScaleNet Network

Our ScaleNet is a lightweight network with four convolution layers. The first convolution layer takes the output of the last encoder layer as input. Each of the first three convolution layers is followed by ELU [3] activation function. The output of the last convolution layer is used to compute a global mean which is followed by a SoftPlus [16] activation function to output a global scale factor. The architecture of our ScaleNet network is explained in Tab. 3.

1.3. Model Complexity

It must be noted that only MDE model and ScaleNet are used during inference, while PoseNet is only used during training. The total number of parameters of our model are $\approx$ 15.6 million to which ScaleNet network's contribution is only $\approx$ 1.33 million. As pointed out in the main paper, PackNet-SfM [7] uses 128 million ($\approx$ 8 times more) parameters, and our model outperforms PackNet-SfM in almost all the metrics. Therefore, with complex network architecture, the performance of MDE model trained with our method can improve further.

Depth Decoder						
layer	**k**	**s**	**chns**	**res**	**input**	**activation**
upconv5	3	1	256	32	econv5	ELU
iconv5	3	1	256	16	↑upconv5, econv4	ELU
upconv4	3	1	128	16	iconv5	ELU
iconv4	3	1	128	8	↑upconv4, econv3	ELU
disp4	3	1	1	1	iconv4	Sigmoid
upconv3	3	1	64	8	iconv4	ELU
iconv3	3	1	64	4	↑upconv3, econv2	ELU
disp3	3	1	1	1	iconv3	Sigmoid
upconv2	3	1	32	4	iconv3	ELU
iconv2	3	1	32	2	↑upconv2, econv1	ELU
disp2	3	1	1	1	iconv2	Sigmoid
upconv1	3	1	16	2	iconv2	ELU
iconv1	3	1	16	1	↑upconv1	ELU
disp1	3	1	1	1	iconv1	Sigmoid

Table 1. **Depth network architecture. k** is the kernel size, **s** is the stride, **chns** is the number of output channels for each layer, **res** is the downscaling factor for each layer relative to the input image, and **input** corresponds to the input of each layer where ↑ is a $2\times$ nearest-neighbor upsampling of the layer.

Pose Decoder						
layer	**k**	**s**	**chns**	**res**	**input**	**activation**
pconv0	1	1	256	32	econv5	ReLU
pconv1	3	1	256	32	pconv0	ReLU
pconv2	3	1	256	32	pconv1	ReLU
pconv3	1	1	6	32	pconv3	-

Table 2. **Pose network architecture.** Abbreviations have same meaning as explained in Tab. 1.

ScaleNet						
layer	**k**	**s**	**chns**	**res**	**input**	**activation**
sconv0	1	1	256	32	econv5	ELU
sconv1	3	1	256	32	sconv0	ELU
sconv2	3	1	256	32	sconv1	ELU
sconv3	3	1	1	32	sconv3	Global Mean & SoftPlus

Table 3. **ScaleNet architecture.** Abbreviations have same meaning as explained in Tab. 1. **Global Mean & SoftPlus** means that we compute mean of the output channel of **sconv3** layer and apply a SoftPlus activation function [16] to get the global scale factor.

2. Ablation Studies: Extra Results and Discussion

In addition to the quantitative results of our ablation studies presented in the main paper, we present detailed results and discussion of our experiments and qualitative comparisons to show that the choices we make in our method results in the best performing MDE model. All our experiments are performed using KITTI Eigen split [4, 5] as monocular videos dataset and VKITTI2 [2] as synthetic dataset. *The experimental results in the ablation study show that the problem at hand is not a trivial task of training a MDE model jointly using self-supervised learning on monocular videos and pixel-wise depth regression task on synthetic dataset.*

First, we clearly define different experiments in our ablation study:

1. **Baseline SS**: This is the baseline model trained using self-supervised training on monocular videos dataset.
2. **Baseline Syn**: This is the baseline model trained on synthetic dataset using a pixel-wise regression loss.
3. **Joint (Naïve)**: This model is trained jointly on monocular videos and synthetic datasets with no-pretraining.
4. **Joint (PT Syn)**: This model is trained jointly on monocular videos and synthetic datasets after synthetic dataset pre-training.
5. **Joint (PT SS)**: This model is trained jointly on monocular videos and synthetic datasets after self-supervised pre-training on monocular videos dataset.
6. **Ours** (L_1): This model is trained using the proposed method till Stage 2. However, for synthetic dataset training in Stage 2, we use a pixel-wise regression loss between $\tilde{d}_{syn\_rel}$ and $\hat{d}_{syn}$ instead of domain specific loss (as described in Eq.(8) in the main paper).
7. **Ours (Grad)**: This model is trained in the same manner as **Ours** (L_1) except that we use a domain specific (i.e., gradient) loss between $\tilde{d}_{syn\_rel}$ and $\hat{d}_{syn}$ which results in best performance.
8. **Ours**: This is the **Ours (Grad)** model trained using the proposed method till Stage 3 and it performs scale-aware (absolute) depth estimation.

As our pixel-wise regression loss for synthetic dataset in Experiment 2-5, we use both, a trivial L_1-norm error based loss as well as the scale-invariant mean squared error based loss (introduced by Eigen *et al.* [4], we denote it by SiLog). SiLog constitutes both L_2-norm term and a scale-invariant term. We use SiLog loss in our experiments to quantitatively justify the Scale Alignment module in Stage 2 of our training method. The results of our experiments are tabulated in Tab. 4, whereas Fig. 1 shows the qualitative comparison of our experiments.

Please Note: In the main paper, we displayed results of **Joint (PT SS)** model trained using L_1 loss on synthetic dataset, whereas the other experiments, viz., **Baseline Syn**, **Joint (Naïve)** and **Joint (PT Syn)** were trained using SiLog loss on synthetic dataset. It was an error from our side. However, here we provide well categorized and consolidated results of all experiments trained with both types of pixel-wise regression losses.

In Tab. 4 we compare methods which estimate relative depth and scale-aware (absolute) depth estimation methods separately. Since **Joint (PT SS)** is pre-trained on real world

data using self-supervision, we additionally evaluate it in the relative depth estimation category to check its performance after joint training.

First, it is clear that all trivial training combinations (i.e., **Joint (Naïve)**, **Joint (PT SS)** and **Joint (PT Syn)**) employing a L_1-norm error based loss function on synthetic dataset perform poorly in both relative as well as absolute depth estimation categories. Instead, **Baseline Syn** performs relatively better compared to these experiments. This shows that competing loss functions during joint training (Section 3.2 in the main paper) also play a significant role apart from synthetic dataset domain bias.

Second, we see that after employing SiLog loss, the performance of **Joint (Naïve)**, **Joint (PT SS)** and **Joint (PT Syn)** improves significantly, while the performance of **Baseline Syn** remains same. It again establishes the above point: Employing SiLog loss, which has a scale-invariant term, reduces the contention between two training losses during joint training and improves the performance. In **Baseline Syn** experiment, there are no conflicting loss functions (i.e., no conflict between self-supervised and synthetic dataset training losses). Therefore, its performance remains almost the same with both L_1 and SiLog.

Third, we also see that after employing SiLog loss, the performance of **Joint (PT SS)** improves significantly more (80% improvement in Abs Rel error) than **Joint (Naïve)**, **Joint (PT Syn)**. In fact, the relative depth estimation performance of **Joint (PT SS)** becomes acceptable after the introduction of SiLog loss on the synthetic dataset. This shows the benefit of pre-training on real world monocular videos using self-supervision; the model learns feature representations specific to real world data during pre-training. During joint training, feature representations do not change significantly, mainly when synthetic dataset training loss is scale-invariant (or atleast has a scale-invariant term in the case of SiLog). **Joint (Naïve)** and **Joint (PT Syn)** which are not pre-trained on real world data, do not show the same kind of performance improvement as **Joint (PT SS)**.

From the above observations and discussions, it can be inferred why the model trained with the proposed method till Stage 2, i.e., **Ours (Grad)** results in the best performance. It employs a Scale Alignment module which does not lead to the contention of losses during joint training. Additionally, it employs a domain specific (i.e., gradient) loss function, which improves performance.

Finally, **Ours** model that is trained till stage 3 of our method is the only model which outputs accurate scale-aware depth. None of the trivial training combinations lead to a model whose absolute depth performance comes closer to **Ours**.

Fig. 1 also shows the qualitative comparison of the results of our experiments. It can be easily inferred that the results of **Ours** are superior compared to the results of other models. The other models underperform because of domain bias, unstable joint training, or both. On the other hand, our approach of disentangling the task of relative depth estimation with qualitative depth attributes from the scale-aware depth estimation task results in the best performance.

3. ScaleNet: Additional Experiments

We conduct additional experiments to test the robustness and domain bias of ScaleNet. In Tab. 5 we include a baseline comparison with respect to using a fixed scale, i.e., the mean scaling ratio between prediction and ground-truth derived from the synthetic data. It can be seen that ScaleNet performs better than using a fixed scaling ratio (FixSyn).

To analyze the domain bias, we train ScaleNet using KITTI training samples with ground-truth (method δ-Real in Tab. 5). ScaleNet trained on synthetic data performs almost similar to or better than δ-Real. Our intuitive understanding is that scale prediction uses the size and position of objects in the scene, similar in both synthetic (VKITTI) and real (KITTI) due to the same camera parameters.

4. Additional Evaluation: Make3D Dataset

Make3D [13] dataset is used to evaluate the generalization capability of MDE models. The dataset is only used for evaluation. It contains images from open urban areas and comes with low-resolution ground-truth depth maps acquired using a 3D scanner.

Tab. 6 shows the quantitative comparison of our method on Make3D dataset. The Make3D dataset is only used for evaluation to compare the generalization performance of MDE models. It can be observed that our relative depth has 4% better Abs Rel error, 16.1% better Sq Rel error, and 6.8% better RMSE compared to Monodepth2 [6] (second best in relative depth). Compared to scale-aware methods, our model has 27% better Sq Rel, and 9% better RMSE than SharinGAN [12]. The state-of-the-art results on Make3D show that the MDE trained with our method has good generalization capability.

Fig. 2 shows the qualitative comparison of our model against Monodepth2 [6]. The depth maps generated by our model are accurate and have sharp boundaries and smooth depth variations, which are the traits acquired from synthetic dataset training.

5. Extra Qualitative Comparisons

5.1. Depth Map Comparisons

We have included additional qualitative results in Fig. 3 to show that the proposed method generates visually more accurate, sharp, and smooth depth maps compared to other methods, which have blur boundaries, holes in reflective surfaces, and missing or bleeding depth for thin objects

Experiment	Syn Loss	Abs Rel	Sq Rel	RMSE	RMSE$_{log}$	$\delta_{1.25}$	$\delta_{1.25^2}$	$\delta_{1.25^3}$
				Relative				
Baseline SS	-	0.115	0.882	4.701	0.190	0.879	0.961	0.982
Joint (PT SS)	L_1	0.430	8.967	12.203	0.462	0.481	0.729	0.854
Joint (PT SS)	SiLog	0.109	0.748	4.456	0.183	0.882	0.965	**0.984**
Ours (L_1)	-	0.106	0.713	4.369	0.181	0.888	**0.966**	**0.984**
Ours (Grad)	-	**0.103**	**0.654**	**4.300**	**0.178**	**0.891**	**0.966**	**0.984**
				Absolute				
Baseline Syn	L_1	0.209	1.828	6.897	0.318	0.664	0.851	0.932
Joint (Naïve)	L_1	0.775	9.631	15.356	1.623	0.001	0.004	0.015
Joint (PT SS)	L_1	0.941	14.147	18.408	3.094	0.001	0.002	0.003
Joint (PT Syn)	L_1	0.900	12.326	16.908	2.618	0.000	0.001	0.002
Baseline Syn	SiLog	0.200	1.588	6.853	0.323	0.663	0.855	0.933
Joint (Naïve)	SiLog	0.548	5.147	11.391	0.833	0.007	0.029	0.146
Joint (PT SS)	SiLog	0.183	1.037	5.365	0.263	0.703	0.936	0.976
Joint (PT Syn)	SiLog	0.588	5.820	12.041	0.930	0.005	0.018	0.080
Ours	-	**0.109**	**0.702**	**4.409**	**0.185**	**0.876**	**0.962**	**0.984**

Table 4. **Ablation study results.** Quantitative results of our ablation study to demonstrate the efficacy of our method.

Method	Abs Rel	Sq Rel	RMSE	RMSE$_{log}$	$\delta_{1.25}$	$\delta_{1.25^2}$	$\delta_{1.25^3}$
FixSyn	0.164	0.897	4.573	0.207	0.831	0.959	0.985
δ-Real	**0.104**	**0.685**	4.468	0.186	**0.880**	**0.962**	0.983
Ours	0.109	0.702	**4.409**	**0.185**	0.876	**0.962**	**0.984**

Table 5. ScaleNet: Additional Experiments

Method	Data	Abs Rel	Sq Rel	RMSE
Zhou [20]	M	0.383	5.321	10.470
Monodepth2 [6]	M	0.330	3.453	7.253
Ours (relative)	M + V	**0.317**	**2.894**	**6.756**
AdaDepth [10]	M+D+V (DA)	0.647	12.341	11.567
Atapour [1]	D+V (DA)	0.423	9.343	9.002
T^2Net [19]	M+V (DA)	0.508	6.589	8.935
GASDA [17]	S+V (DA)	0.403	6.709	10.424
SharinGAN [12]	S+V (DA)	0.377	4.900	8.388
Ours (absolute)	M + V	**0.370**	**3.572**	**7.632**

Table 6. **Make3D results.** Quantitative comparison with existing state-of-the-art on Make3D dataset. Symbols and styles have same meaning as described in Table 1 in the main paper.

(e.g., poles). We additionally include a qualitative comparison with PackNet-SfM [7] which is absent in the main paper due to space constraints.

5.2. Point-cloud Comparisons

We have also included additional 3D point cloud comparisons in Fig. 4 to show that the proposed method generates accurate scale-aware depth, which preserves scene geometry and shapes of objects much better than state-of-the-art methods.

6. Human-centric Synthetic Depth Dataset

In the main paper, we demonstrated the practical usefulness of our method in developing an MDE model for the task of applying DSLR like synthetic depth-of-field effect [15], popular as Portrait Mode on smartphones. We also showed representative images from our in-house synthetic depth dataset that we created using computer graphics software Blender [9]. In Fig. 5, we show additional representative images along with corresponding dense depth maps. We generate 3000 synthetic RGB-D pairs for our work, and we will make the dataset available to the community. We expect it to be helpful in human-centric vision research, such as portrait effect, human segmentation, and human depth estimation [14].

References

[1] Amir Atapour Abarghouei and Toby P. Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In *CVPR*, 2018. 4

[2] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2, 2020. 2

[3] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). In *ICLR*, 2016. 1

[4] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NeurIPS*, 2014. 2

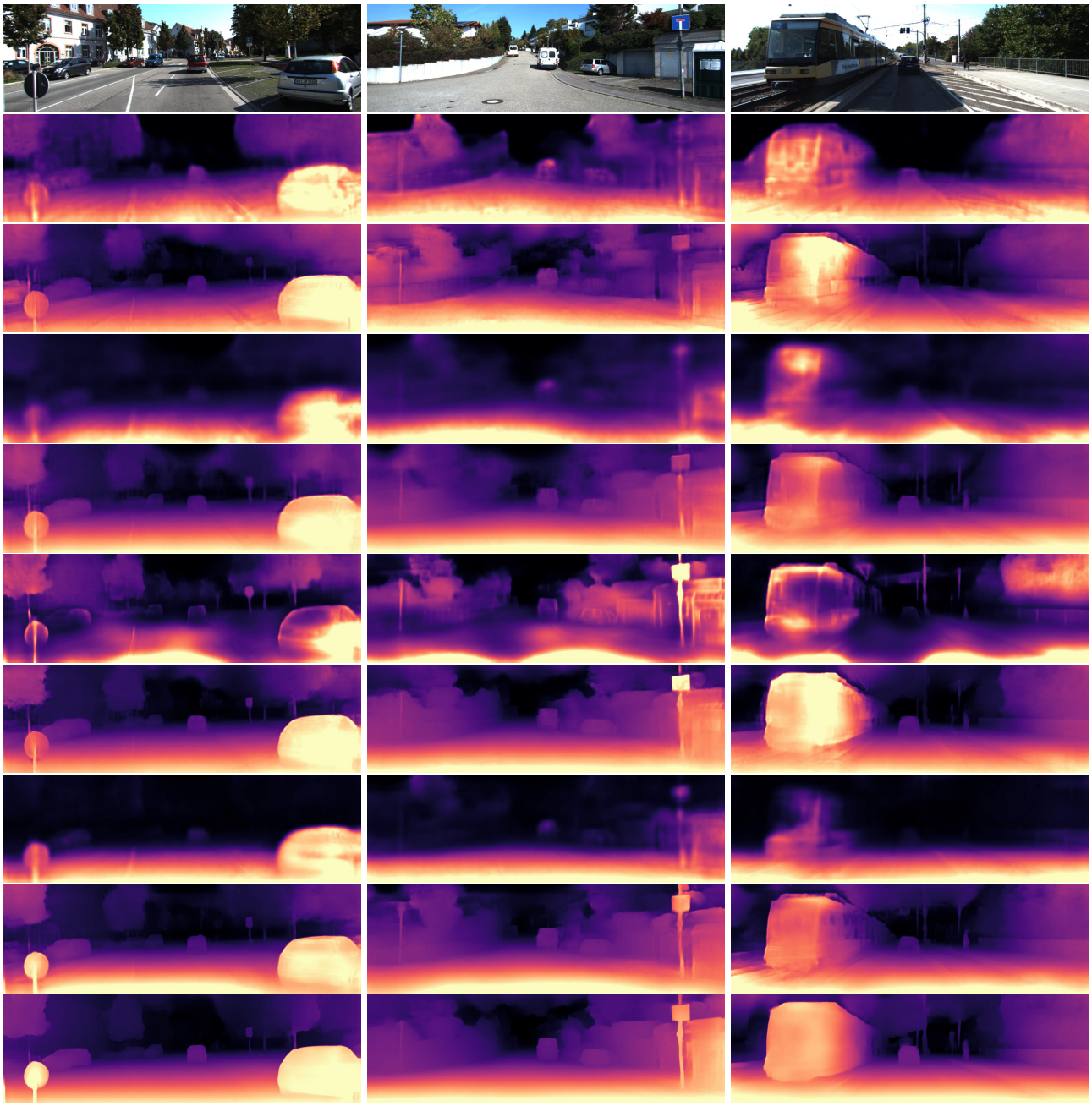


Figure 1. **Ablation study results.** Qualitative comparison of results of different experiments in our ablation study. The images are in following order (row wise): Input, Baseline Syn L_1, Baseline Syn SiLog, Joint (Naïve) L_1, Joint (Naïve) SiLog, Joint (PT SS) L_1, Joint (PT SS) SiLog, Joint (PT Syn) L_1, Joint (PT Syn) SiLog and Ours. As it can be seen, the proposed method (last row) leads to edge-consistent depth estimation, smooth depth variations, no bleeding object edges and no holes within objects.

[5] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *Int. J. Robotics Res.*, 2013. 2

[6] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, 2019. 1, 3, 4, 6, 7, 8

[7] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *CVPR*, 2020. 1, 4, 7, 8

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

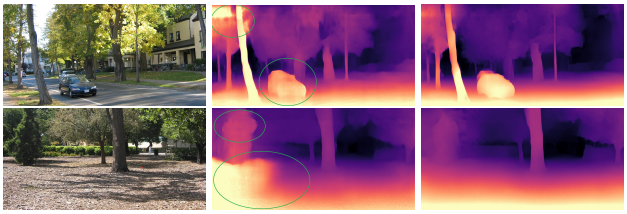


Figure 2. **Make3D results.** Qualitative comparison with [6]. First column shows input images, second column shows results of [6] and third column shows our results. Zoom-in and pay attention to regions marked in green color. Our results have smooth depth variations and very less bleeding at object edges.

[9] Roland Hess. *Blender Foundations: The Essential Guide to Learning Blender 2.6*. Focal Press, 2010. 4

[10] Jogendra Nath Kundu, Phani Krishna Uppala, Anuj Pahuja, and R Venkatesh Babu. Adadepth: Unsupervised content congruent adaptation for depth estimation. In *CVPR*, 2018. 4

[11] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016. 1

[12] Koutilya PNVR, Hao Zhou, and David Jacobs. Sharingan: Combining synthetic and real data for unsupervised geometry estimation. In *CVPR*, 2020. 3, 4, 7

[13] Ashutosh Saxena, Min Sun, and Andrew Y. Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009. 3

[14] Feitong Tan, Hao Zhu, Zhaopeng Cui, Siyu Zhu, Marc Pollefeys, and Ping Tan. Self-supervised human depth estimation from monocular videos. In *CVPR*, 2020. 4

[15] Neal Wadhwa, Rahul Garg, David E. Jacobs, Bryan E. Feldman, Nori Kanazawa, Robert Carroll, Yair Movshovitz-Attias, Jonathan T. Barron, Yael Pritch, and Marc Levoy. Synthetic depth-of-field with a single-camera mobile phone. *ACM Trans. Graph.*, 2018. 4

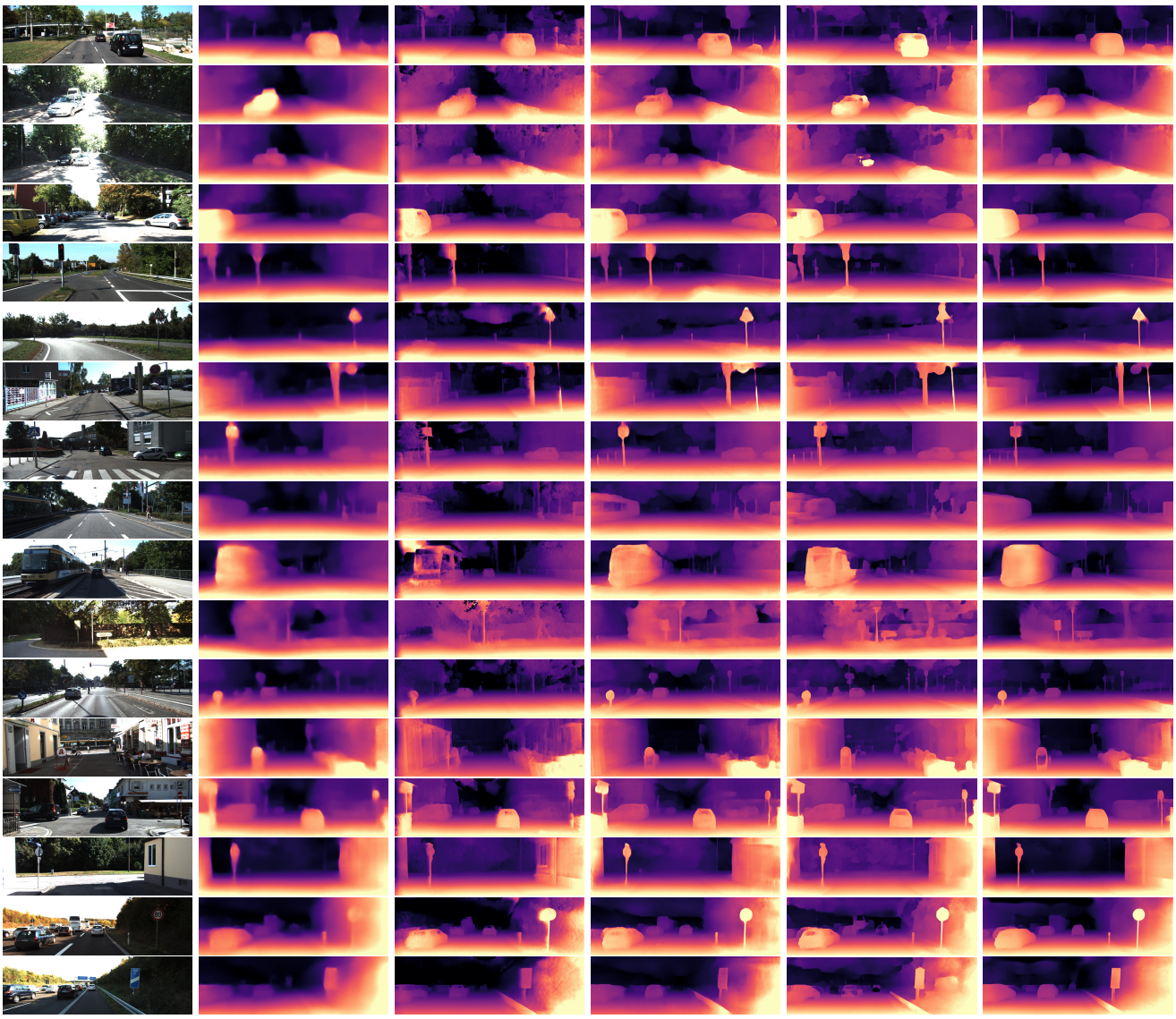
[16] Huizhen Zhao, Fuxian Liu, Longyue Li, and Chang Luo. A novel softplus linear unit for deep convolutional neural networks. *Applied Intelligence*, 2018. 1, 2

[17] Shanshan Zhao, Huan Fu, Mingming Gong, and Dacheng Tao. Geometry-aware symmetric domain adaptation for monocular depth estimation. In *CVPR*, 2019. 4

[18] Wang Zhao, Shaohui Liu, Yezhi Shu, and Yong-Jin Liu. Towards better generalization: Joint depth-pose learning without posenet. In *CVPR*, 2020. 7

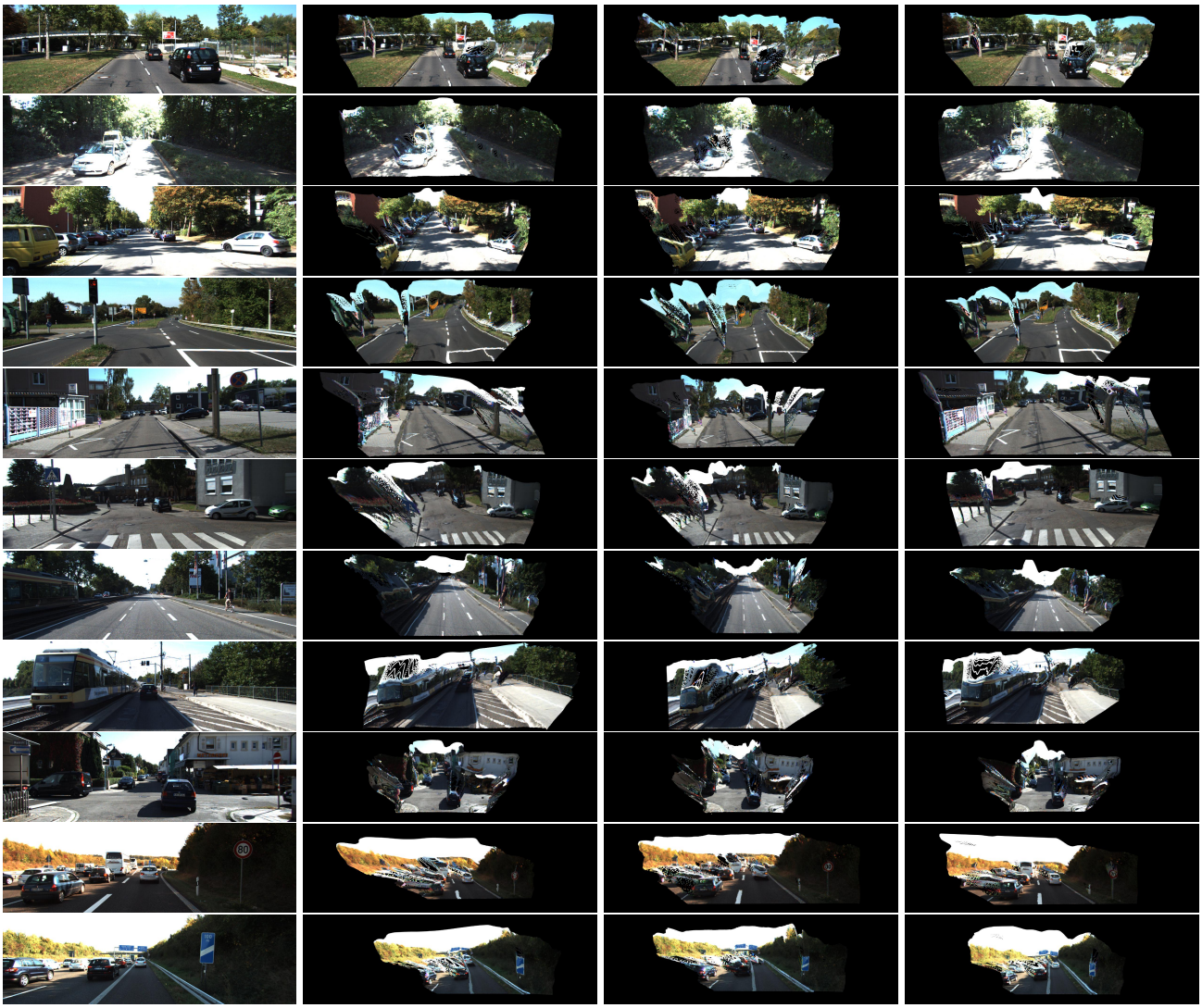
[19] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. T2net: Synthetic-to-realistic translation for solving single-image depth estimation tasks. In *ECCV*, 2018. 4

[20] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017. 4



(a) Input (b) Zhao [18] (c) SharinGAN [12] (d) Monodepth2 [6] (e) PackNet-SfM [7] (f) Ours

Figure 3. **KITTI results.** Additional qualitative comparison with state-of-the-art, we also include results of PackNet-SfM [7] which are absent in the main paper. The proposed method leads to edge-consistent depth estimation, smooth depth variations, no bleeding object edges and no holes within objects (particularly on reflective surfaces).



(a) Input (b) Monodepth2 [6] (c) PackNet-SfM [7] (d) Ours

Figure 4. **KITTI results.** Additional qualitative comparison with state-of-the-art, the reconstructed 3D point cloud using our accurate scale-aware depth preserves scene geometry and shapes of objects much better than state-of-the-art methods.

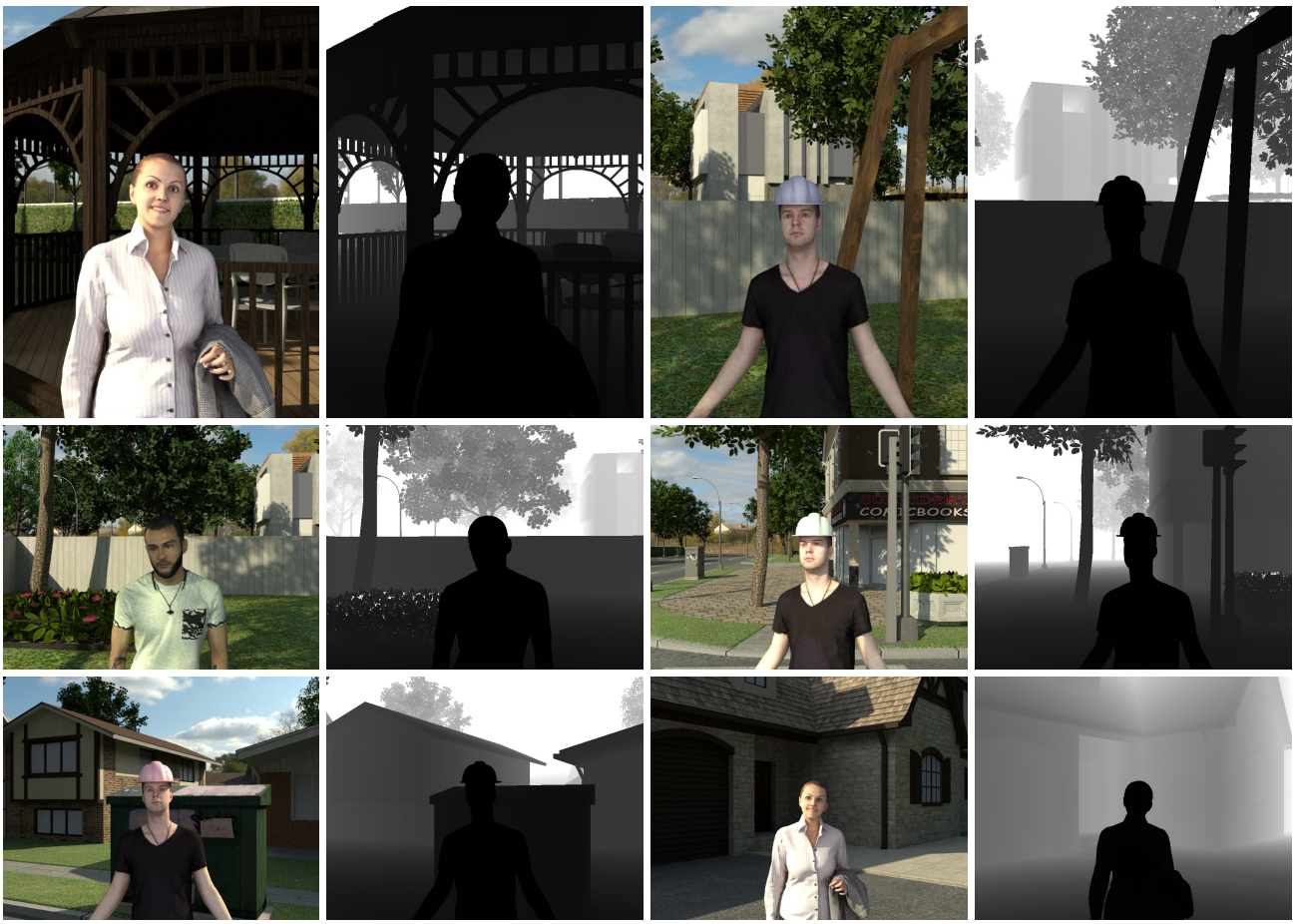


Figure 5. Representative input images from our in-house synthetic depth dataset.