Camera Height Doesn't Change: Unsupervised Training for Metric Monocular Road-Scene Depth Estimation Supplementary Material

Genki Kinoshita and Ko Nishino

Graduate School of Informatics, Kyoto University, Kyoto, Japan
gkinoshita@vision.ist.i.kyoto-u.ac.jp, kon@i.kyoto-u.ac.jp
https://vision.ist.i.kyoto-u.ac.jp/

A Details of Training on the Mixed Datasets

A.1 Dataset Preparation

We use the images from the front camera in Argoverse2 [21], Lyft [12], A2D2 [9], and DDAD [11] for training. As mentioned in Sec. 6.1 since the focal lengths are essential for metric MDE, we resize the image and align the focal lengths to 1000 px. In order to facilitate the training process, we crop the images to 832×512 resolution. These resizing and cropping are also done for evaluation on the KITTI Eigen split [7].

To make the training easier and more efficient, we eliminate frames where the camera is stationary since training with these frames greatly degrades the inference performance [10]. We achieve this elimination by looking at pairs of successive frames and simply checking the number of pixels with large differences in normalized intensities. Images with a small number of pixels with large intensity changes are deemed stationary and discarded from the training dataset. Since Argoverse2 is captured at a higher frame rate than the other datasets, we treat every other frame as successive frames.

For optimization during training, FUMET requires a sufficient number of images captured with the same camera height. We select such sequences with at least 2,500 frames. At the end of each training epoch, we optimize each camera height individually. In total, we use 235,341 images for training.

A.2 Training Settings

We train Monodepth2 [10] with ResNet50 on FUMET with the same batch size and initial learning rate as used during training on the KITTI dataset (Sec. 6.1). Considering the gap of training samples between KITTI and the mixed dataset, we set $\tau_{\text{mid}} = 4$ and train for 8 epochs while reducing the learning rate by half every 3 epochs. For training on the mixed dataset and KITTI, we train for 8 epochs with $\tau_{\text{mid}} = 3$ and reduce the learning rate by half every 2 epochs.

Table 1: Comparison of FUMET trained with ground-truth camera intrinsics versus the one trained with recovered values with COLMAP. Each model is trained and evaluated on KITTI [7,8] (640×192 resolution). FUMET can train a model with recovered camera intrinsics, achieving accuracy comparable to that trained with ground-truth ones.

	Error (↓)				Accuracy (↑)		
Intrinsics	AbsRel	SqRel	RMSE	$\text{RMSE}_{\log}$	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
GT	**0.108**	0.785	4.736	**0.195**	**0.871**	**0.958**	**0.981**
COLMAP	0.110	**0.774**	**4.721**	0.197	0.867	**0.958**	**0.981**

B Camera Intrinsics Recovered with COLMAP

Similar to previous self-supervised MDE methods, FUMET requires camera intrinsics to compute the photometric error in training. In the following section, we demonstrate that FUMET successfully trains a model with intrinsics recovered with COLMAP [18] on the KITTI dataset [8]. In Sec. B.2, we also verify that FUMET can leverage in-the-wild videos for training by using COLMAP.

B.1 Training on KITTI with Recovered Camera Intrinsics

We run COLMAP with the pinhole camera assumptions on KITTI sequences containing 500 to 800 frames and use the median of the computed intrinsics for training. We train Monodepth2 [10] (ResNet50) on FUMET under the same training settings as described in Sec. 6.1. The results on KITTI (Tab. 1) show that thanks to the accuracy of COLMAP, the model trained with the recovered intrinsics achieves comparable accuracy to that trained with the ground-truth values.

B.2 Training with In-the-wild Videos

We download two YouTube videos [1, 2] (73,100 frames) and run COLMAP to recover the intrinsics. Since these videos are too long to use all the frames for COLMAP, we manually extract consecutive 300 frames from uncrowded scenes and run COLMAP individually with the pinhole camera assumptions. We train Monodepth2 with FUMET on the dataset composed of the mixed dataset (Argoverse2 [21], Lyft [12], A2D2 [9], and DDAD [11]), KITTI and the YouTube videos. The batch size and the initial learning rate are the same as those used in Sec. 6.1. For the same reason as described in Sec. A.2, we train for 6 epochs with $\tau_{\text{mid}} = 2$, reducing the learning rate by half every 2 epochs.

The results on KITTI shown in Tab. 2 demonstrate that adding in-the-wild videos to the training dataset increases the dataset diversity and improves the depth accuracy. This suggests the significance of being able to train without sensor measurements.

Table 2: FUMET can leverage YouTube videos for training and lead to slightly improved accuracy. The results are evaluated on KITTI [7] (832×512 resolution). In the Dataset column, M, K, and Y represent the mixed dataset, KITTI, and YouTube, respectively.

Dataset	Error (↓)				Accuracy (↑)		
	AbsRel	SqRel	RMSE	$\text{RMSE}_{\log}$	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
M + K	**0.082**	0.611	4.307	0.149	**0.923**	0.976	**0.989**
M + K + Y	0.084	**0.608**	**4.264**	**0.148**	**0.923**	**0.977**	**0.989**

C Assumption of Vehicle Presence

In contrast to inference time, FUMET leverages observed vehicles and creates pseudo scale supervision for training. However, it does not create pseudo supervision for each frame independently but instead aggregates the scale information into the camera height and optimizes it across the sequence. This allows FUMET to create scale supervision from frames observing vehicles and provide scale supervisory signals even for frames without observed vehicles. In fact, as shown in Fig. 1, although the training dataset of KITTI contains many frames where vehicles are not observed or are scarce, FUMET can train with high depth accuracy.

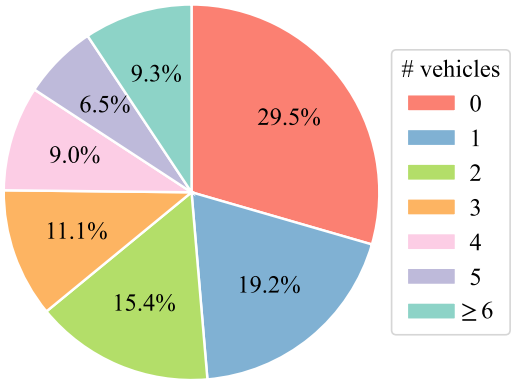

Fig. 1: The proportion of frames with the number of observed cars in the training dataset of KITTI [8]. There is no or only one observed car in nearly half of the frames in the dataset,

Table 3: Comparison with ZoeDepth [3] trained on the mixed dataset with LiDAR depth. The results are evaluated on KITTI [7] (832×512 resolution). FUMET slightly outperforms ZoeDepth in terms of depth accuracy even though FUMET does not use ground-truth depth maps.

Method	Supervision	Error (↓)				Accuracy (↑)		
		AbsRel	SqRel	RMSE	$RMSE_{log}$	δ<1.25	δ<1.25^2	δ<1.25^3
ZoeDepth [3]	GT	0.120	**0.743**	5.118	0.182	0.870	**0.964**	**0.988**
FUMET	-	**0.113**	0.916	**5.009**	**0.181**	**0.883**	0.961	0.983

D Comparison to a Supervised Method

Our key contribution is a self-supervision cue for metric depth reconstruction. As such, direct comparison with supervised methods (*e.g.*, ZoeDepth [3]) would be unfair. That said, we can compare with supervised methods trained on our mixed dataset using LiDAR depth. We use ZoeDepth as a representative method for this comparison. We select the architecture most similar to our default model (ResNet50) from its official implementation in terms of the number of parameters and the architecture type, *i.e.*, ConvNet, and skip pre-training performed in the original paper for fairness.

The results on KITTI (Tab. 3) show that FUMET slightly outperforms ZoeDepth. This suggests that it can achieve accuracy comparable to supervised methods. This is likely because the richer self-supervision from the reconstruction loss (Eq. 3) forces the model to recognize scene contexts effectively, more so than sparse LiDAR depth, and enables metric scale inference from them without overfitting.

E Implementation Details of LSP

We implement LSP with PyTorch [16]. We train it for 15 epochs with a batch size of 32 and warm-up the first 5 epochs. We use AdamW [14] optimizer with a learning rate of 5.0×10^{-4} linearly decreased to 4.0×10^{-4} and a weight decay of 0.05. As a backbone network, we employ ConvNeXt V2 [22] pre-trained with ImageNet22K [17] and concatenate two linear layers to it. As a training dataset, we use DVM-CAR [13], which contains 1,451,784 vehicle mask images from 899 UK market car models and their dimensions. All images are resized to 300×300 resolution. We use 80% of the data for training and others for validation. We adopt multiple data augmentations mentioned in Sec. 5 to make the model robust to the domain gap. We use L1 loss for each dimension as the objective function.

F Ablation Studies of LSP

As specified in Sec. 5 in the main paper, during training of LSP, we perform beam/instance occlusion and truncation augmentations illustrated in Fig. 2

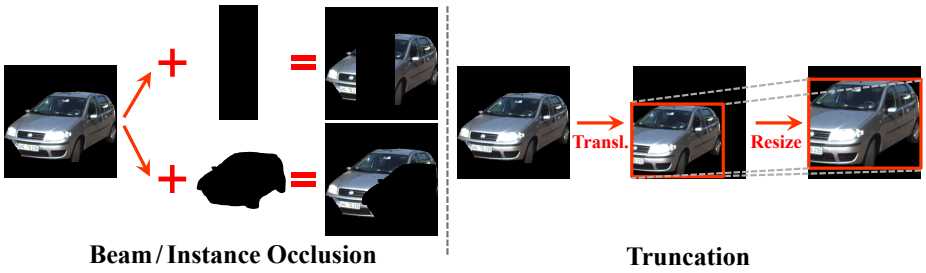

Fig. 2: Visualization of beam/instance occlusion and truncation augmentations for training LSP.

Table 4: Ablation of augmentations for LSP on KITTI [5] and Cityscapes [6]. We evaluate each predicted dimension (height/width/length) on the absolute relative error. Both beam/instance occlusion and truncation augmentation improve the prediction accuracy.

beam occ.	inst. occ.	trunc.	KITTI	Cityscapes
			0.081 / 0.159 / 0.174	0.088 / 0.065 / 0.082
	✓	✓	0.071 / 0.162 / 0.179	**0.080** / 0.069 / 0.080
✓		✓	0.074 / 0.150 / 0.172	0.083 / 0.065 / 0.078
✓	✓		0.079 / **0.136** / 0.165	0.084 / **0.057** / **0.077**
✓	✓	✓	**0.062** / 0.147 / **0.145**	**0.080** / 0.064 / **0.077**

along with common color jittering and blur. The beam occlusion simulates objects occluded by poles including traffic lights and trees. Their segmentation mask is divided into more than two regions. The instance occlusion augmentation is for objects occluded by the other things including vehicles and people. Truncation augmentation is for objects that are partially cropped out of images. In Tab. 4, we ablate the effectiveness of these augmentations. The results show that each augmentation contributes to improving the prediction accuracy both on the KITTI and the Cityscapes datasets, *i.e.*, they enhance the robustness to any input image.

G Additional Quantitative Results on KITTI

In the main paper, we trained the multiple networks with FUMET on the KITTI dataset [8] and evaluated them on the KITTI Eigen split [7] which is the most widely used, comparing the weakly-supervised methods. We also evaluate the same models with the improved ground-truth depth maps for KITTI introduced in [19]. These improved depth maps are composed of 652 out of 697 test frames in the Eigen split. The results shown in Tab. 5 demonstrate that the models trained with FUMET still surpass weakly-supervised methods, both with and without the median scaling, as well as when evaluating on the KITTI Eigen

Table 5: Evaluation on KITTI improved ground truth [19] (640×192 resolution). The models trained with FUMET outperform weakly-supervised methods in both results, with and without the median scaling [26].

Method	Supervision	Scaling	Error (↓)				Accuracy (↑)		
			AbsRel	SqRel	RMSE	$RMSE_{log}$	δ<1.25	δ<1.25^2	δ<1.25^3
G2S [4]	GPS	-	0.111	0.791	4.523	0.168	0.869	0.967	0.989
PackNet-SfM [11]	V	-	0.105	0.656	4.098	0.156	0.886	0.971	0.991
Wagstaff *et al.* [20]	CamH	-	0.097	0.636	4.281	0.153	0.887	0.975	0.993
VADepth [23]	CamH	-	0.091	0.555	3.871	0.134	**0.913**	0.983	0.995
DynaDepth [25]	IMU+V+G	-	0.092	**0.477**	3.756	0.135	0.906	0.983	**0.996**
Ours	-	-	**0.089**	0.506	3.806	0.134	0.908	**0.984**	**0.996**
Lite-Mono [24]+**Ours**	-	-	0.090	0.526	3.767	0.134	0.910	**0.984**	0.995
HR-Depth [15]+**Ours**	-	-	0.096	0.524	3.852	0.139	0.902	0.982	**0.996**
VADepth [23]+**Ours**	-	-	0.101	0.547	**3.675**	**0.133**	0.910	**0.984**	**0.996**
G2S [4]	GPS	✓	0.088	0.544	3.968	0.137	0.913	0.981	0.995
PackNet-SfM [11]	V	✓	0.100	0.606	3.943	0.145	0.900	0.977	0.993
Wagstaff *et al.* [20]	CamH	✓	0.095	0.602	4.145	0.146	0.902	0.978	0.994
VADepth [23]	CamH	✓	0.080	0.518	3.745	0.123	0.933	0.986	0.996
DynaDepth [25]	IMU+V+G	✓	0.084	0.448	3.761	0.130	0.917	0.984	0.996
Ours	-	✓	0.081	0.455	3.675	0.125	0.927	0.986	0.996
Lite-Mono [24]+**Ours**	-	✓	0.083	0.478	3.632	0.126	0.926	0.986	0.996
HR-Depth [15]+**Ours**	-	✓	0.084	0.461	3.719	0.129	0.922	0.985	0.996
VADepth [23]+**Ours**	-	✓	**0.077**	**0.426**	**3.455**	**0.116**	**0.937**	**0.989**	**0.997**

split. It should be emphasized that VADepth [23] trained with FUMET (row 9 and 18) outperforms the same network supervised with the measured camera height (row 4 and 13). This proves that the pseudo camera height obtained via the camera height optimization is more precise than the measured one.

H Additional Qualitative Results on KITTI

We provide additional results of FUMET in diverse scenes on the KITTI dataset. As shown in Fig. 3, FUMET can predict more accurate depth maps than the weakly-supervised methods, especially in road and object regions.

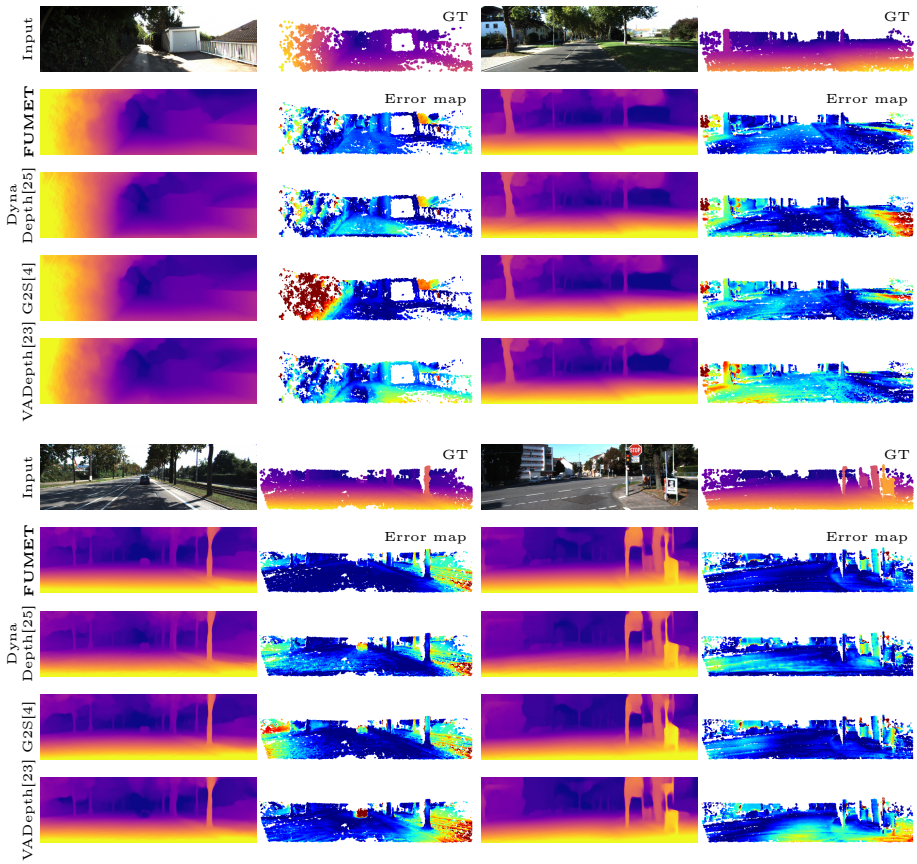

Fig. 3: Additional qualitative comparison on KITTI. We compare estimated depth maps of Monodepth2 [10] trained with FUMET to the ones of weakly-supervised methods: G2S [4], DynaDepth [25], and VADepth [23]. In error maps, the larger depth errors are represented in red, and smaller ones are depicted in blue. The results show that FUMET can predict more accurate depth maps in various scenes, compared with the weakly-supervised methods.

References

1. 4K Drive: 4K DRIVE USA [Raleigh] North Carolina NC GoPro Hero 9 driving (Jul 2021), https://www.youtube.com/watch?v=RpCcS7vyECs
2. 4K Drive: 4K DRIVE Punta del ESTE 2022 Uruguay UY 4k video GoProHero 9 (Jan 2022), https://www.youtube.com/watch?v=dgUi7xUpqCA
3. Bhat, S.F., Birkl, R., Wofk, D., Wonka, P., Müller, M.: Zoedepth: Zero-shot Transfer by Combining Relative and Metric Depth. arXiv preprint arXiv:2302.12288 (2023)
4. Chawla, H., Varma, A., Arani, E., Zonooz, B.: Multimodal Scale Consistency and Awareness for Monocular Self-Supervised Depth Estimation. In: ICRA. pp. 5140–5146 (2021)
5. Chen, X., Kundu, K., Zhu, Y., Berneshawi, A.G., Ma, H., Fidler, S., Urtasun, R.: 3D Object Proposals for Accurate Object Class Detection. Adv. Neural Inform. Process. Syst. **28** (2015)
6. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The Cityscapes Dataset for Semantic Urban Scene Understanding. In: CVPR (2016)
7. Eigen, D., Fergus, R.: Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture. In: ICCV. pp. 3213–3223 (2015)
8. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision Meets Robotics: The KITTI Dataset. International Journal of Robotics Research (IJRR) **32**(11), 1231—-1237 (2013). https://doi.org/10.1177/0278364913491297
9. Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A.S., Hauswald, L., Pham, V.H., Mühlegg, M., Dorn, S., Fernandez, T., Jänicke, M., Mirashi, S., Savani, C., Sturm, M., Vorobiov, O., Oelker, M., Garreis, S., Schuberth, P.: A2D2: Audi Autonomous Driving Dataset. CoRR **abs/2004.06320** (2020)
10. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J.: Digging Into Self-Supervised Monocular Depth Estimation. In: ICCV. pp. 3827–3837 (2019)
11. Guizilini, V., Ambrus, R., Pillai, S., Raventos, A., Gaidon, A.: 3D Packing for Self-Supervised Monocular Depth Estimation. In: CVPR. pp. 2482–2491 (2020)
12. Houston, J., Zuidhof, G., Bergamini, L., Ye, Y., Chen, L., Jain, A., Omari, S., Iglovikov, V., Ondruska, P.: One Thousand and One Hours: Self-Driving Motion Prediction Dataset. In: Conference on Robot Learning. pp. 409–418 (2021)
13. Huang, J., Chen, B., Luo, L., Yue, S., Ounis, I.: DVM-CAR: A Large-Scale Automotive Dataset for Visual Marketing Research and Applications. In: International Conference on Big Data. pp. 4140–4147 (2022)
14. Loshchilov, I., Hutter, F.: Decoupled Weight Decay Regularization. In: ICLR (2019)
15. Lyu, X., Liu, L., Wang, M., Kong, X., Liu, L., Liu, Y., Chen, X., Yuan, Y.: HR-Depth: High Resolution Self-Supervised Monocular Depth Estimation. In: AAAI. vol. 35, pp. 2294–2301 (2021)
16. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: Adv. Neural Inform. Process. Syst. pp. 8026—-8037 (2019)
17. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet: A Large Scale Visual Recognition Challenge. IJCV **115**, 211–252 (2015)

18. Schonberger, J.L., Frahm, J.M.: Structure-From-Motion Revisited. In: CVPR. pp. 4104–4113 (2016)
19. Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A.: Sparsity Invariant CNNs. In: 3DV. pp. 11–20 (2017)
20. Wagstaff, B., Kelly, J.: Self-Supervised Scale Recovery for Monocular Depth and Egomotion Estimation. In: IROS. pp. 2620–2627 (2021)
21. Wilson, B., Qi, W., Agarwal, T., Lambert, J., Singh, J., Khandelwal, S., Pan, B., Kumar, R., Hartnett, A., Pontes, J.K., Ramanan, D., Carr, P., Hays, J.: Argoverse 2: Next Generation Datasets for Self-driving Perception and Forecasting. In: NeurIPS Datasets and Benchmarks (2021)
22. Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I.S., Xie, S.: ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders. In: CVPR. pp. 16133–16142 (2023)
23. Xiang, J., Wang, Y., An, L., Liu, H., Wang, Z., Liu, J.: Visual Attention-Based Self-Supervised Absolute Depth Estimation Using Geometric Priors in Autonomous Driving. IEEE Robotics and Automation Letters **7**(4), 11998–12005 (2022). https://doi.org/10.1109/LRA.2022.3210298
24. Zhang, N., Nex, F., Vosselman, G., Kerle, N.: Lite-Mono: A Lightweight CNN and Transformer Architecture for Self-Supervised Monocular Depth Estimation. In: CVPR. pp. 18537–18546 (2023)
25. Zhang, S., Zhang, J., Tao, D.: Towards Scale-Aware, Robust, and Generalizable Unsupervised Monocular Depth Estimation by Integrating IMU Motion Dynamics. In: ECCV. pp. 143–160 (2022)
26. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised Learning of Depth and Ego-Motion from Video. In: CVPR. pp. 6612–6619 (2017)