GEDepth: Ground Embedding for Monocular Depth Estimation

SUPPLEMENTARY MATERIAL

Xiaodong Yang* Zhuang Ma* Zhiyu Ji* Zhe Ren
QCraft

A. Training Configurations

In the paper, we have extensively evaluated GEDepth with four representative networks including DepthFormer [20], PixelFormer [1], BinsFormer [21], and BTS [16], which represent the state-of-the-art methods of monocular depth estimation in both Transformers and CNNs. For fair comparisons, we follow the original training configuration of each method when training GEDepth.

- **DepthFormer** and **PixelFormer:** We set the bath size as 16 on 8 GPUs and the initial learning rate as 1e-4. We use the cosine annealing learning rate for 38.4K iterations and apply the linear learning rate warm-up strategy for the first 30% iterations.
- **BinsFormer:** We set the batch size to 16 on 8 GPUs and the initial learning rate to 1e-4. We adopt the one-cycle learning rate for 38.4K iterations and use the linear learning rate warm-up for the first 30% iterations.
- **BTS:** We set the batch size as 64 on 8 GPUs and the initial learning rate as 1e-4. We utilize the cosine annealing learning rate scheduler for 24 epochs without using the warm-up strategy.

AdamW is used as the optimizer for all networks above.

B. Where to Embed Ground Depth

As illustrated in Figure 2, our approach originally embeds the ground depth in the input. Here we evaluate the performance by embedding the ground depth in the encoder as an alternative. Table 8 shows that embedding in the encoder also improves over the baseline DepthFormer, but is inferior to the original embedding.

Method	Abs Rel ↓	RMSE ↓	SILog ↓
DepthFormer	0.052	2.133	7.210
GE-Vanilla (encoder)	0.050	2.071	7.074
GE-Vanilla (input)	0.049	2.063	6.983
GE-Adaptive (encoder)	0.049	2.070	7.072
GE-Adaptive (input)	0.048	2.050	6.982

Table 8. Comparison of where to embed ground depth in GEDepth.

*Authors contributed equally

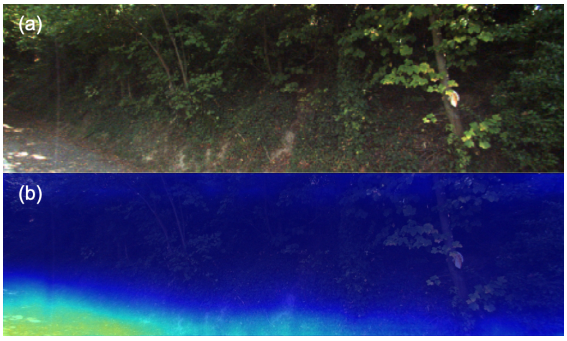

Figure 11. Visualization of the predicted ground attention map (b) in the scene (a) where ground is barely present.

Method	Abs Rel ↓	RMSE ↓
DepthFormer	0.140	1.018
GE-Vanilla	0.073	0.543
GE-Adaptive	**0.068**	**0.502**

Table 9. Comparison of baseline (DepthFormer) with GEDepth-Vanilla and GEDepth-Adaptive in the scenario where ground is barely present as shown in Figure 11.

C. Scenes Lacking of Ground

Although ground is ubiquitous in the camera images captured by autonomous driving vehicles, here we probe into how GEDepth performs in the rare case where ground is barely present in the test set of KITTI. As shown in Figure 11, our approach is still able to predict reasonably accurate ground attention map. Table 9 reports that the overall result in this scene degrades compared to the common ones where ground is apparently present, while our approach still improves over the baseline.

D. Hyper-Parameter

We next evaluate how our approach behaves with different values of λ_{cls}, the classification loss weight for ground slope learning in Equation (10) of the paper. As shown in Table 10, our approach is overall robust to the values of this hyper-parameter in a reasonable range ($\lambda_{\text{cls}} = 0.10$ is the default value used in our experiments).

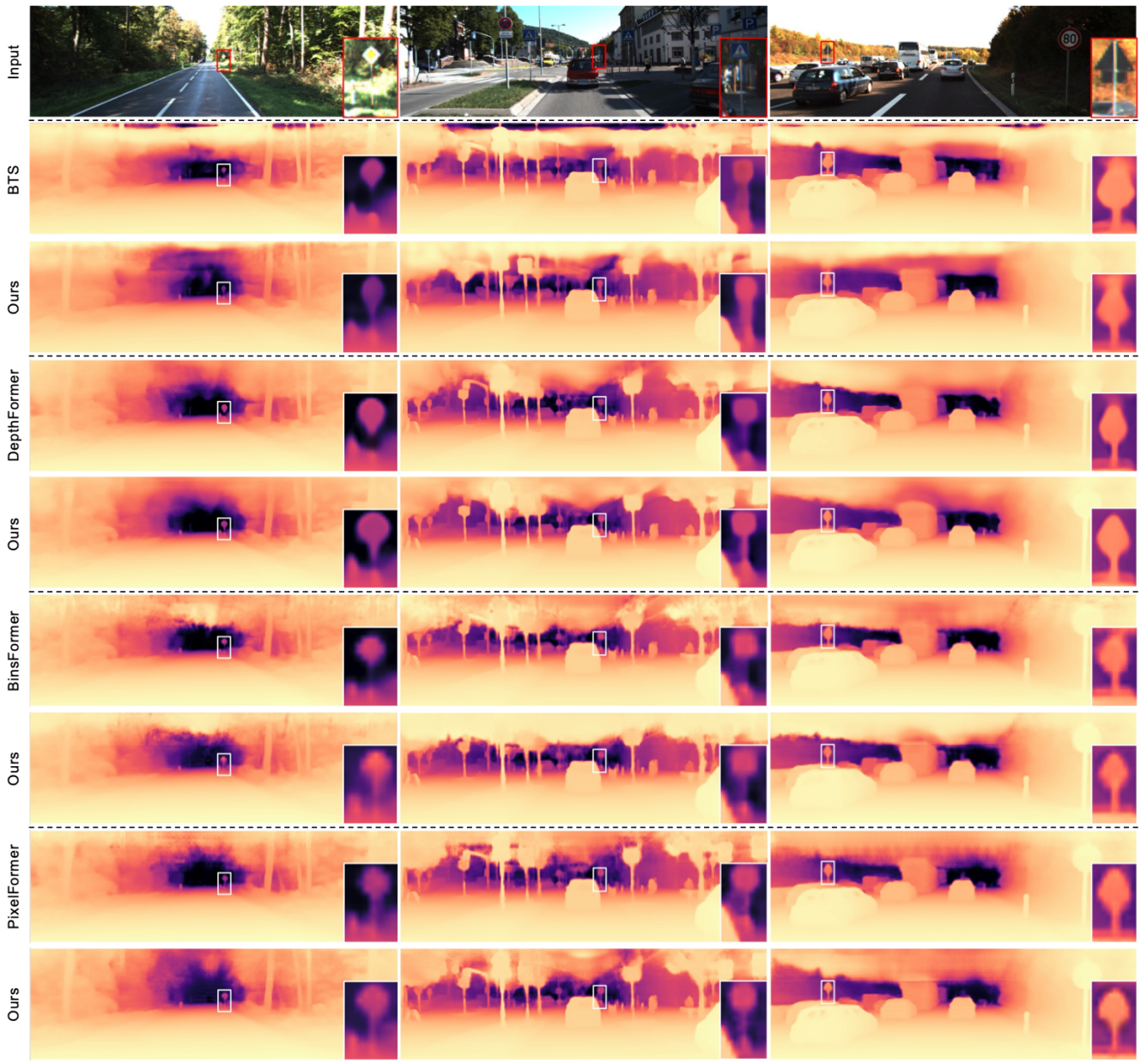

Figure 12. Comparison of the depth prediction results by the state-of-the-art methods and our approach on three scenes of KITTI.

λ_{cls}	Abs Rel	Sq Rel	RMSE	RMSE-log
0.05	0.049	0.147	2.064	0.077
0.08	0.049	0.143	2.045	0.076
0.10	0.048	0.142	2.050	0.076
0.12	0.049	0.144	2.062	0.077
0.15	0.049	0.145	2.064	0.077

Table 10. Evaluation of the hyper-parameter λ_{cls} in Equation (10).

E. Qualitative Results

Figure 12 provides the qualitative results of our approach (GEDepth-Adaptive) and the corresponding state-of-the-art methods on the test set of KITTI. As can be seen in this figure, our approach produces sharper depth prediction, and we observe that the improvements on the distant objects and fine structures are more evident.