iDisc: Internal Discretization for Monocular Depth Estimation Supplementary Material

Luigi Piccinelli Christos Sakaridis Fisher Yu
Computer Vision Lab, ETH Zürich

A. Results

Outdoor zero-shot. We present in Table 7 the results of models pre-trained on KITTI Eigen-split [5] and tested on Argoverse [4] and DDAD [7] test split we proposed in this work. The zero-shot results clearly demonstrate how every model tends to perform poorly when trained on KITTI and tested on a different domain. However, iDisc is able to almost double the performance when directly trained on either Argoverse or DDAD. This suggests that KITTI is not indicative of generalization performance. This investigation leads us to realize the need for more diversity in the outdoor scenario. We address the problem by proposing new dataset splits to train and validate models on. Fig. 16 shows how models fail completely when predicting unseen scenario, *e.g.*, graffiti on a flat wall. In addition, Fig. 17 displays how models under-scale depth when testing on domains with a typical object size, *i.e.*, DDAD in the United States, larger than that of the training set, *i.e.*, KITTI in Germany.

KITTI [6] benchmark. Table 8 clearly shows the compelling performance of iDisc on the official KITTI private test set. We show the results of the latest published methods only. The table is from the official KITTI leaderboard.

IDRs collapse. We argue that our model is able to avoid over-clustering when performing the adaptive partitioning in AFP step. Over-clustering is the phenomenon occurring when the number of partitions enforced is more than the underlying true one. The ID module is able to avoid over-clustering by degenerating some IDRs onto others, thus not introducing any detrimental partition of the feature space. Degeneration of the same IDR is visible in Fig. 6.

Attention depth planes. Fig. 7 shows three IDRs (each row shows a specific IDR, as in main paper figures) at the middle resolution. The top two rows support the "speculation" on iDisc's ability to still capture depth planes.

Computational complexity. We provide the analysis of the components in Table 10. Removing MSDA increases throughput to 20fps, with only a slight loss in performance. Note that our implementation is not fully optimized for performance. NeWCRF [18] uses the same backbone but more parameters and similar throughput to iDisc without MSDA.

Table 7. **Zero-shot testing of models trained on KITTI Eigen-split.** Comparison of performance when methods are trained on KITTI Eigen-split and tested, without further fine-tuning, on the splits of Argoverse and DDAD introduced in this work.

Test set	Method	$\delta_1 \uparrow$	RMS $\downarrow$	A.Rel $\downarrow$	$\text{SI}_{\log} \downarrow$
Argoverse	BTS [10]	0.307	15.98	0.383	51.80
	AdaBins [3]	0.383	17.07	0.350	52.33
	P3Depth [13]	0.277	17.97	0.376	44.09
	NeWCRF [18]	0.311	15.75	0.370	46.77
	Ours	**0.560**	**12.18**	**0.269**	**33.35**
DDAD	BTS [10]	**0.399**	16.19	0.350	40.51
	AdaBins [3]	0.282	18.36	0.433	50.71
	P3Depth [13]	0.397	17.83	**0.330**	39.00
	NeWCRF [18]	0.343	16.76	0.375	44.24
	Ours	0.350	**14.26**	0.367	**29.37**

Table 8. **Results on official KITTI [6] Benchmark.** Comparison of performance of methods trained on KITTI and tested on the official KITTI private test set.

Method	$\text{SI}_{\log}$	Sq.Rel	A.Rel	iRMS
	Lower is better			
PAP [19]	13.08	2.72 %	10.27 %	13.95
P3Depth [13]	12.82	2.53 %	9.92 %	13.71
VNL [16]	12.65	2.46 %	10.15 %	13.02
DORN [17]	11.77	2.23 %	8.78 %	12.98
BTS [10]	11.67	2.21 %	9.04 %	12.23
PWA [11]	11.45	2.30 %	9.05 %	12.32
ViP-DeepLab [14]	10.80	2.19 %	8.94 %	11.77
NeWCRF [18]	10.39	1.83 %	8.37 %	11.03
PixelFormer [1]	10.28	1.82 %	8.16 %	10.84
Ours (iDisc)	**9.89**	**1.77 %**	**8.11 %**	**10.73**

Table 9. **Comparison on NYU with 3D metrics.** F1-score for varying threshold (m) and Chamfer distance (m) on point clouds.

Method	$\text{F1}_{0.05} \uparrow$	$\text{F1}_{0.1} \uparrow$	$\text{F1}_{0.2} \uparrow$	$\text{F1}_{0.3} \uparrow$	$\text{F1}_{0.5} \uparrow$	$\text{F1}_{0.75} \uparrow$	$\text{D}_{\text{Chamfer}} \downarrow$
BTS [10]	24.5	47.0	72.4	84.4	93.6	97.2	0.169
AdaBins [3]	24.0	47.0	73.0	84.7	94.0	97.4	0.163
NeWCRF [18]	25.5	48.6	74.0	85.4	94.4	97.6	0.156
iDisc	**27.8**	**52.0**	**77.0**	**87.8**	**95.5**	**98.1**	**0.131**

B. Ablations

Number of IDRs. We ablate the model with respect to the number of IDRs exploited by iDisc. In particular, we sweep

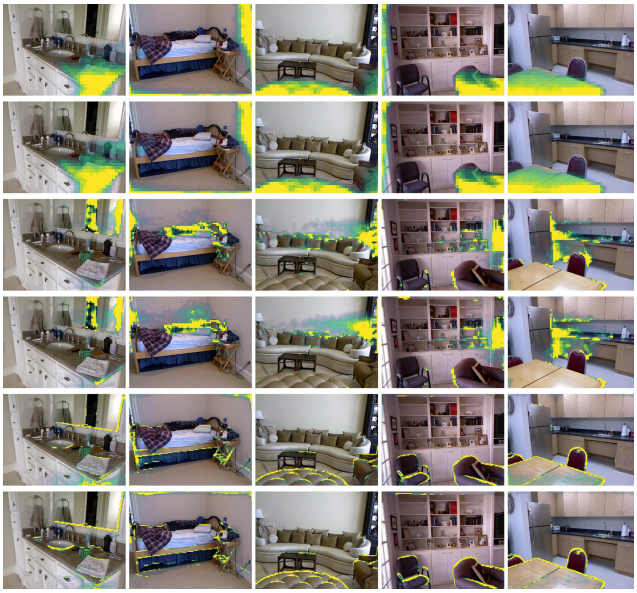

Figure 6. **Examples of attention maps degeneration.** Each pair of rows shows two different IDRs' attention maps, each pair is extracted from a different resolution. Some IDRs degenerate onto other IDRs, avoiding over-partitioning when more IDRs than those needed are utilized to represent the scene.

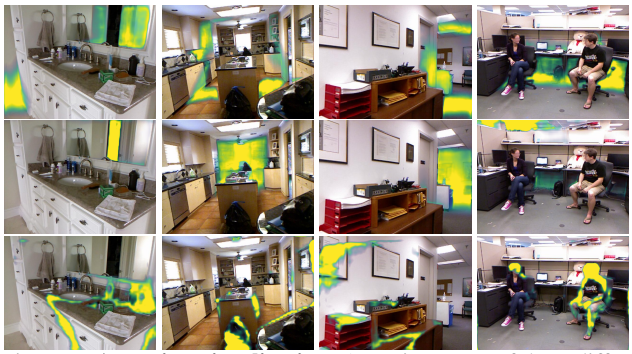

Figure 7. **Attention visualization.** Attention maps of three different IDRs at mid-resolution, on four different images from NYU.

Table 10. **Computational complexity analysis** on an RTX 3090 with input images of size 640×480 and SWin-L backbone.

Component	**Latency** (ms)	**Throughput** (fps)	**Parameters** (M)
Encoder	23.6	42.4	194.9
MSDA	72.8	13.7	2.83
FPN	2.7	370.5	4.11
AFP	12.4	80.7	2.78
ISD	9.6	103.7	4.59
iDisc (w/o MSDA)	48.2	20.7	206.4
iDisc	121.1	8.3	209.2

the number of IDRs between 2 and 128 with a base-two log scale. The black-solid line in Fig. 8 shows the trend of iDisc when ablating the IDRs: the optimum is reached in the interval [8, 32]. When more representations are utilized, we argue that noise is introduced in the bottleneck and the discretization process is not actually enforced. The discretization does not occur since the number of IDRs would be close to the number of feature map elements. On the other hand, 2 or 4 IDRs are already enough to obtain decent results, although not particularly visually appealing. In particular, we speculate that the extreme case of utilizing two IDRs can lead to the model representing the maximum depth with one of the two representations and the minimum one with the other. Therefore, the model is still able to interpolate between the depth interval range. The interpolation occurs thanks to the convex combination, defined by $\mathrm{softmax}$, of maximum and minimum depth. More specifically, $\mathrm{softmax}$ is guided by the similarity between the pixel embeddings and the corresponding depth representations. Thus, the model is virtually able to define the full depth range via the weights of the $\mathrm{softmax}$ convex combination modulated by the pixel embeddings. When utilizing only one representation, the model does not converge, if not to the mean scene depth.

Single resolution in ISD. The dotted-blue line in Fig. 8 shows the trend when only one resolution is processed in the ISD stage of the ID module. In such a configuration, the output of the ID module is directly the depth. Here, no fusion is to be performed between different intermediate representations. One can observe that single-resolution is particularly affected when few IDRs are utilized. We argue that multi-resolution counterparts can compensate for the diminished granularity of internal representation. The compensation stems from combining different facets, *i.e.*, at different resolutions, of the IDRs.

Attention in AFP. The dashed-red line in Fig. 8 shows the performance when standard cross-attention is utilized in AFP, instead of the partition-inducing transposed cross-attention. In this case, a high number of IDRs does not affect performance. Here, the IDRs are additive instead of soft mutually exclusive, *i.e.*, the IDRs from transposed cross-attention. Therefore, utilizing more IDRs is virtually not detrimental.

ID module layers and iterations. Table 11 shows the ablation study on the iterations and layers utilized in the stages of the ID module. We can observe that a higher number of transposed cross-attention, thus of iterative partitioning, has almost no effect on performances, since the partitions have probably converged. On the other hand, when $\mathrm{N_{AFP}}$ is one, results are similar to using only the IDRs priors since the adaptive part is truncated too early. Iterations of ISD stage ($\mathrm{N_{ISD}}$) correspond to the number of cross-attention layers utilized in the last stage of the ID module. iDisc is already able to obtain good results with only one layer, while increasing the layers may lead to overfitting. Nonetheless, Table 12 clearly shows how the input-dependency in the feature partitioning, *i.e.*, $\mathrm{N_{AFP}}$ greater than zero, leads to improved generalization.

C. Network Architecture

Encoder. We show the effectiveness of our method with different encoders, both convolutional and transformer-based

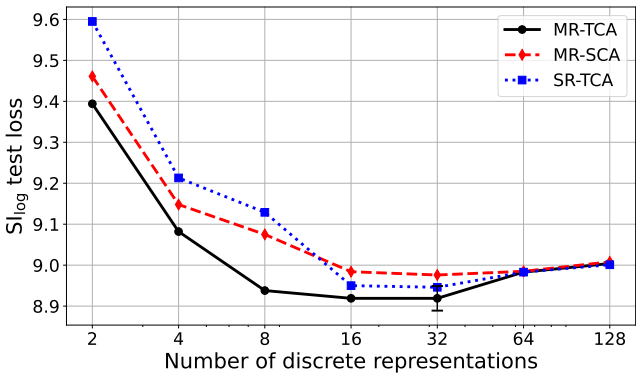

Figure 8. **Ablations on the number of IDRs and ID module's configurations.** MR-TCA: Multi-Resolution and Transposed cross-attention, MR-SCA: Multi-Resolution and Standard cross-attention in AFP, Single-Resolution and Transposed cross-attention. MR-TCA corresponds to the iDisc model. MR-SCA corresponds to using cross-attention instead of cluster-inducing transposed attention. SR-TCA corresponds to having only one intermediate representation, namely the final depth directly. The error bar in correspondence of 32 on the x-axis indicates the standard deviation.

Table 11. **Ablations of ID module iterations.** $\mathrm{N_{AFP}}$: number of iterations in the AFP stage, $\mathrm{N_{ISD}}$: number of cross-attention layers in ISD stage. The last row corresponds to the architecture utilized for all other experiments.

	$\mathrm{N_{AFP}}$	$\mathrm{N_{ISD}}$	$\boldsymbol{\delta_1}\uparrow$	**RMS** $\downarrow$	**A.Rel** $\downarrow$
1	2	1	0.938	0.314	0.086
2	2	3	0.934	0.316	0.088
3	2	4	0.935	0.317	0.089
4	1	2	0.935	0.317	0.087
5	3	2	0.938	0.313	0.086
6	4	2	0.938	0.314	0.086
7	2	2	0.940	0.313	0.086

Table 12. **Test loss for varying $\mathrm{N_{AFP}}$.** The models are trained on NYU and tested on the "Test Dataset".

Test Dataset	$\mathrm{SI_{log}@N_{AFP}}=0$	$\mathrm{SI_{log}@N_{AFP}}=1$	$\mathrm{SI_{log}@N_{AFP}}=2$
NYU	10.43	9.471	8.845
SUN-RGBD	12.76	11.50	10.91
Diode	20.97	18.97	18.11

ones, *e.g.*, ResNet [8], EfficientNet [15] and SWin [12]. However, all of them follow the same structure, where the receptive field of either convolution or windowed attention is increased by decreasing the resolution of the feature maps. The final size of the feature map is 1/32 of the input image. All backbones utilized are originally designed for classification, thus we remove the last 3 layers, *i.e.*, the pooling layer, fully connected layer, and softmax layer. We employ each backbone to generate feature maps of different resolutions, which can be used as skip connections to the decoder.

Multi-scale deformable attention refinement. The feature maps at different resolutions are refined via mutli-scale deformable attention [20]. Deformable attention efficiency relies on attending only a few locations to compute attention for each pixel, instead of having full connectivity likewise standard attention. Deformable attention is also utilized to share information at different resolutions. Each layer is composed of layer normalization [2] (LN), fully connected layers (FC), and Gaussian Error Linear Unit [9] (GeLU).

Decoder. Feature maps at different resolutions are combined via a feature pyramidal network (FPN) which exploits LN, GeLU activations, and convolutional layers with 3×3 kernels. The decoder outputs at different resolutions correspond to the set of pixel embeddings ($\mathcal{P}$).

AFP and ISD. AFP stage is an iterative component, thus weights are shared across layers. One layer comprises transposed cross-attention, LN, GeLU activations, and FC layers: three dedicated layers for key, queries and value tensors, and one layer applied to the attention layer output. The architectural components of the ISD stage are the same as AFP's components, except for the use of standard cross-attention instead of transposed one, and the weights are not shared.

D. Visualizations

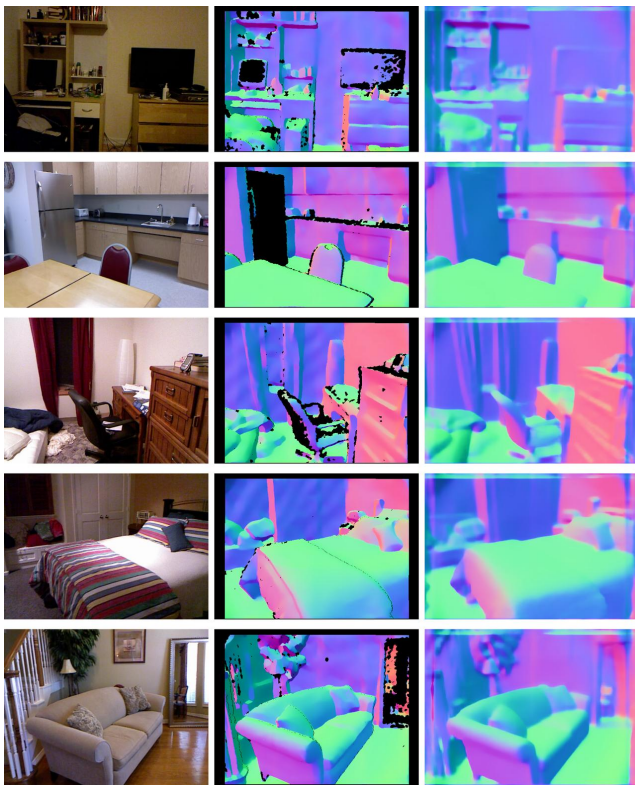

Figure 9. **Qualitative results on NYU for surface normals estimation.** Each row corresponds to one test sample from NYU. The first two columns correspond to the input image and depth GT, respectively. The third column is the predicted normals of the tangent plane for every pixel.

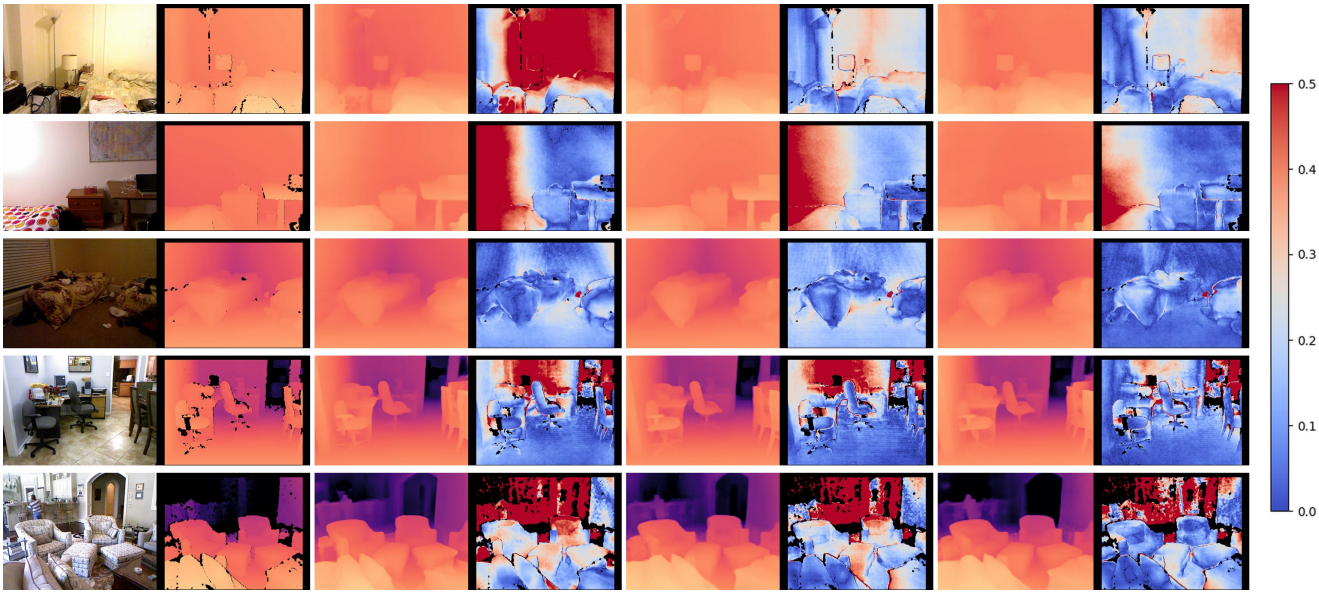

Figure 10. **Qualitative results on NYU.** Each row corresponds to one test sample from NYU. The first two columns correspond to the input image and depth GT, respectively. Each couple afterward corresponds to the pair output depth and error map. Error maps are clipped at 0.5m and the corresponding colormap is *coolwarm*.

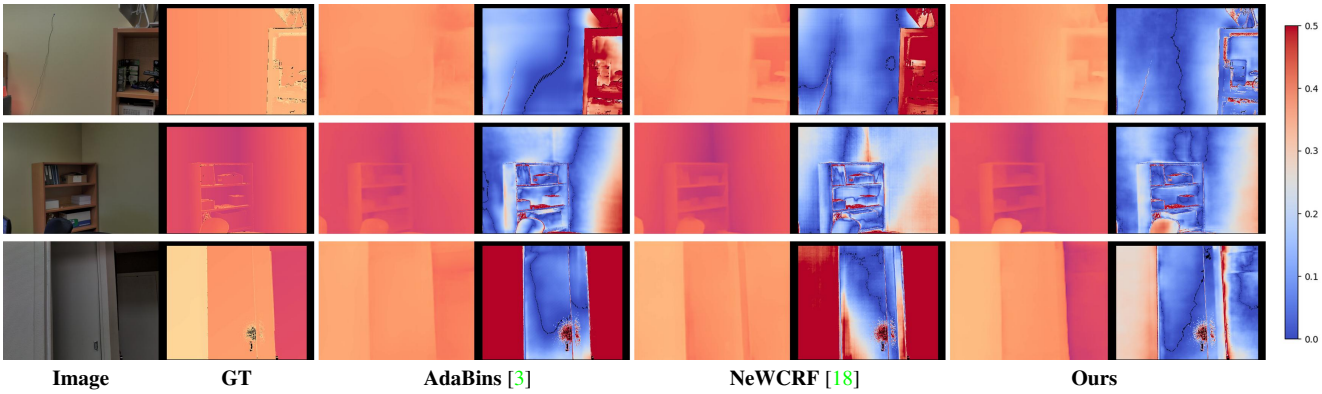

Figure 11. **Qualitative results on Diode.** Each row corresponds to one zero-shot test sample for the model trained on NYU and tested on Diode. The first two columns correspond to the input image and depth GT, respectively. Each subsequent couple corresponds to the pair output depth and error map. Error maps are clipped at 0.5m and the corresponding colormap is *coolwarm*.

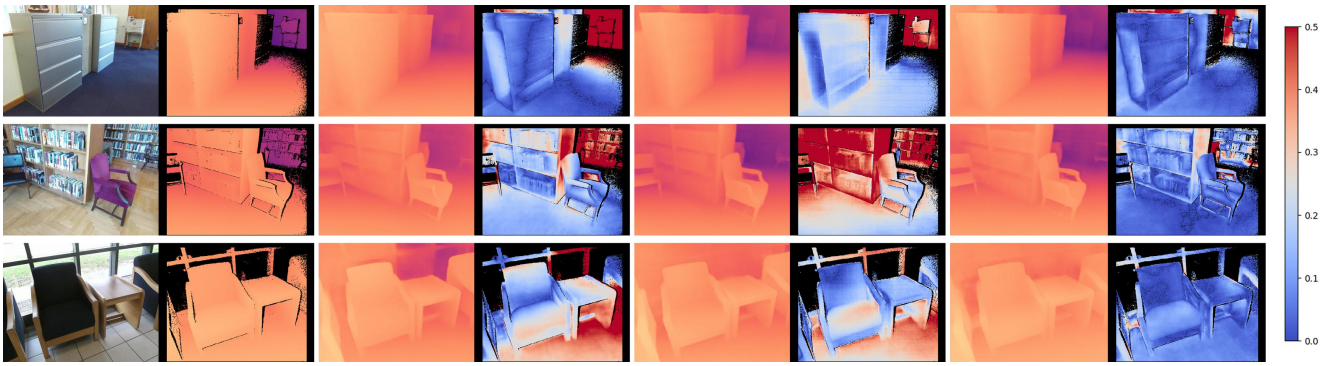

Figure 12. **Qualitative results on SUN-RGBD.** Each row corresponds to one zero-shot test sample for the model trained on NYU and tested on SUN-RGBD. The first two columns correspond to the input image and depth GT, respectively. Each subsequent couple corresponds to the pair output depth and error map. Error maps are clipped at 0.5m and the corresponding colormap is *coolwarm*.

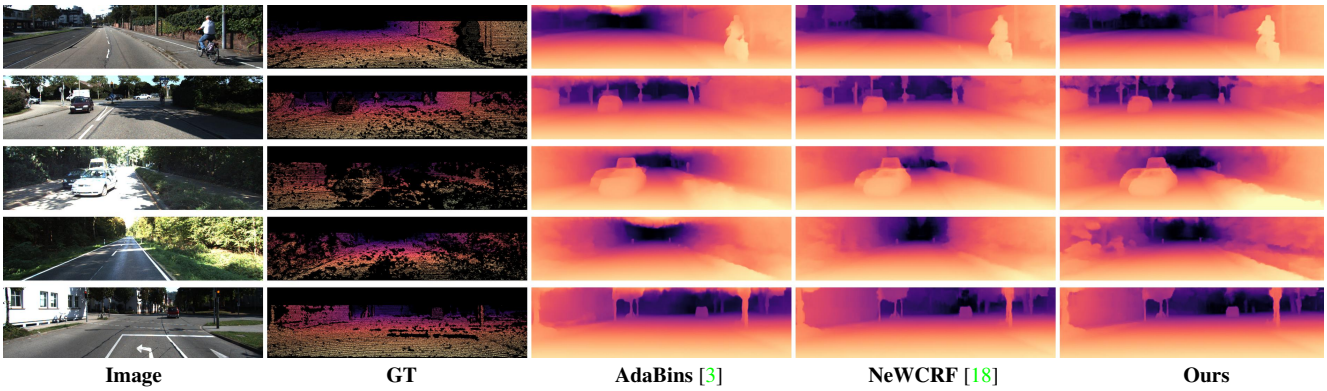

Figure 13. **Qualitative results on KITTI.** Each row corresponds to a test sample from KITTI. The first two columns correspond to the input image and depth GT, respectively. The following columns correspond to the respective models trained on KITTI.

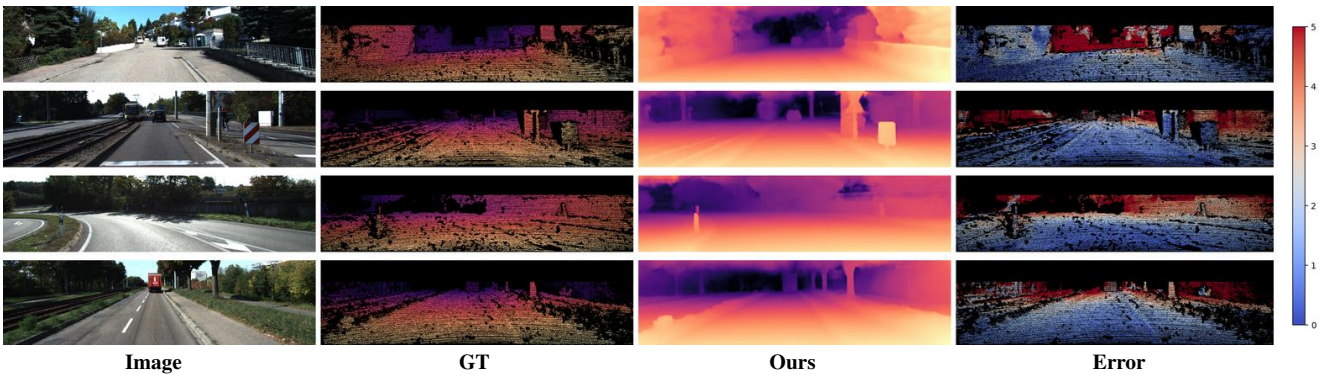

Figure 14. **Failure cases on KITTI.** Each row corresponds to one test sample from KITTI Eigen-split validation set. The examples selected correspond to the four worst samples in terms of absolute error. Error maps are clipped at 5m and the corresponding colormap is *coolwarm*.

Figure 15. **Attention maps on KITTI for three different IDRs.** Each row presents the attention map of a specific IDR for four test images. Each IDR focuses on a specific high-level concept. The first two rows pertain to IDR at the lowest resolution while the last corresponds to the highest resolution. Best viewed on a screen and zoomed in.

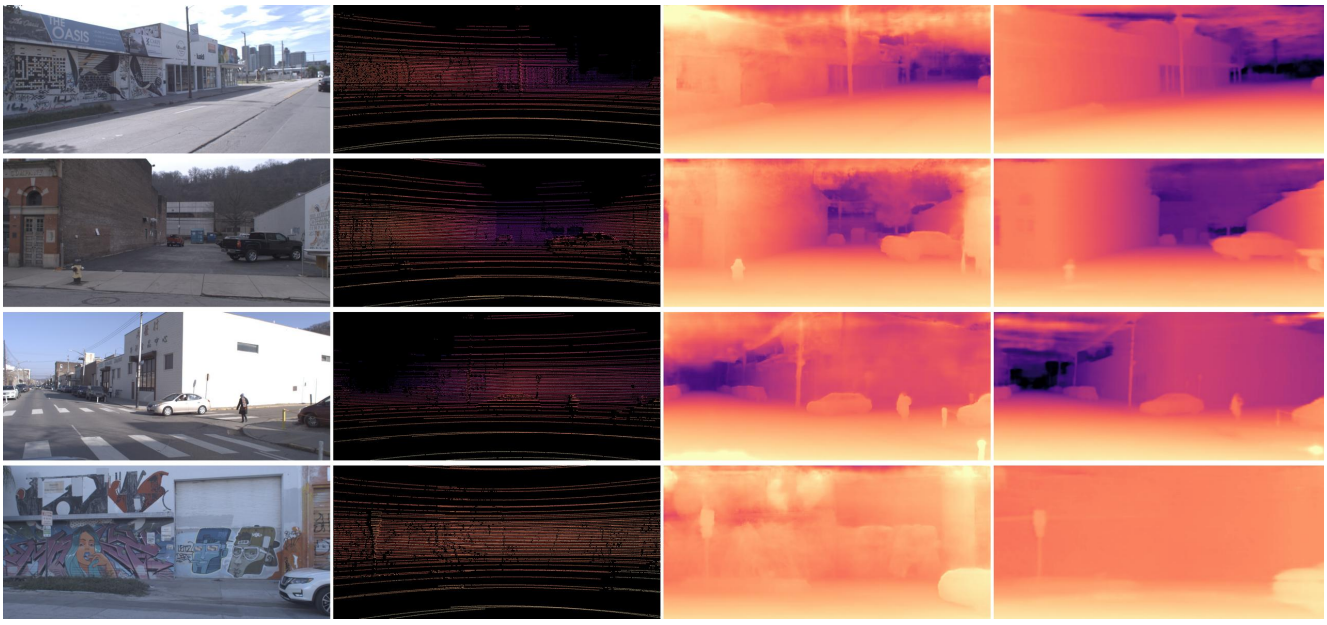

Figure 16. **Qualitative results on Argoverse.** Each row corresponds to one zero-shot test sample from Argoverse. The third column displays the prediction of iDisc trained on KITTI and tested on Argoverse, while the fourth column corresponds to a model trained and tested on Argoverse.

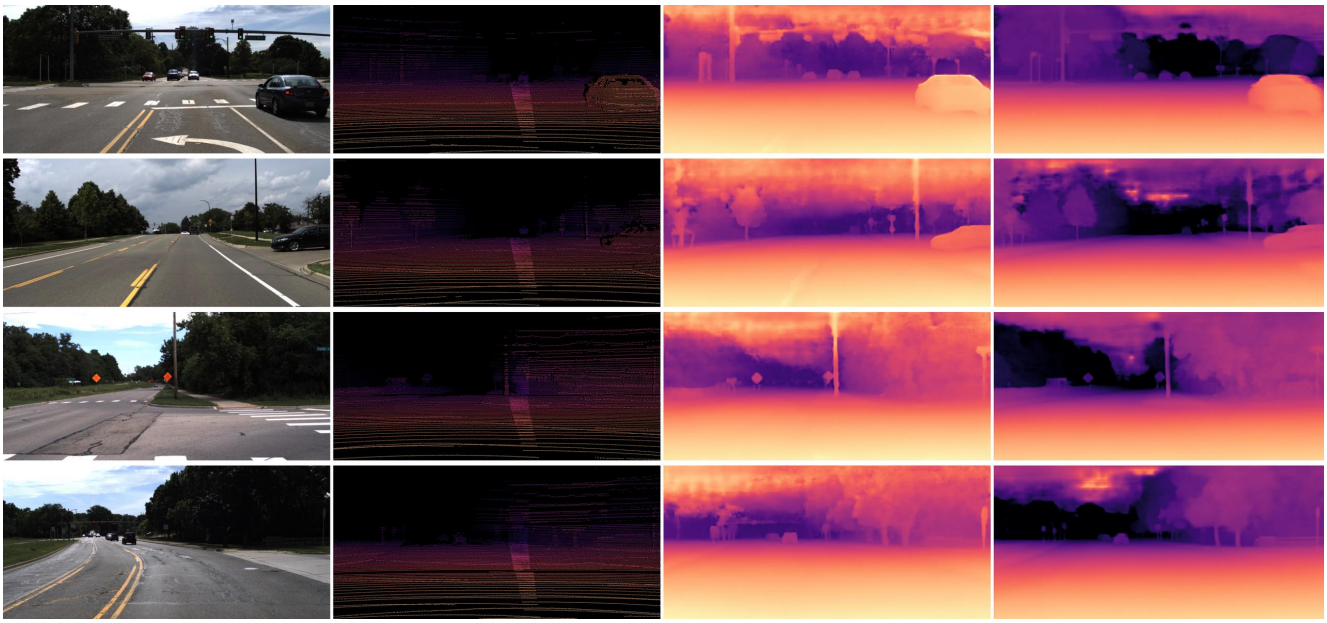

Figure 17. **Qualitative results on DDAD.** Each row corresponds to one zero-shot test sample from DDAD. The third column displays the prediction of iDisc trained on KITTI and tested on DDAD, while the fourth corresponds column to a model trained and tested on DDAD.

References

[1] Ashutosh Agarwal and Chetan Arora. Attention attention everywhere: Monocular depth prediction with skip attention. *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5850–5859, 2022. 1

[2] Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv e-prints*, abs/1607.06450, 2016. 3

[3] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 4008–4017, 11 2020. 1, 4, 5

[4] Ming Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:8740–8749, 11 2019. 1

[5] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in Neural Information Processing Systems*, 3:2366–2374, 6 2014. 1

[6] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 1

[7] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-December:770–778, 12 2015. 3

[9] Dan Hendrycks and Kevin Gimpel. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *arXiv e-prints*, abs/1606.08415, 2016. 3

[10] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv e-prints*, abs/1907.10326, 7 2019. 1

[11] Sihaeng Lee, Janghyeon Lee, Byungju Kim, Eojindl Yi, and Junmo Kim. Patch-wise attention network for monocular depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1873–1881, May 2021. 1

[12] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *Proceedings of the IEEE International Conference on Computer Vision*, pages 9992–10002, 3 2021. 3

[13] Vaishakh Patil, Christos Sakaridis, Alexander Liniger, and Luc Van Gool. P3Depth: Monocular depth estimation with a piecewise planarity prior. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1600–1611. IEEE, 2022. 1

[14] Siyuan Qiao, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3997–4008, 2021. 1

[15] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:10691–10700, 5 2019. 3

[16] Wei Yin, Yifan Liu, Chunhua Shen, and Youliang Yan. Enforcing geometric constraints of virtual normal for depth prediction. *Proceedings of the IEEE International Conference on Computer Vision*, pages 5683–5692, 7 2019. 1

[17] Zehao Yu, Jia Zheng, Dongze Lian, Zihan Zhou, and Shenghua Gao. Single-image piece-wise planar 3d reconstruction via associative embedding. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:1029–1037, 2 2019. 1

[18] Weihao Yuan, Xiaodong Gu, Zuozhuo Dai, Siyu Zhu, and Ping Tan. Neural window fully-connected crfs for monocular depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 3906–3915. IEEE, 2022. 1, 4, 5

[19] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR*, pages 4101–4110, 6 2019. 1

[20] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *9th International Conference on Learning Representations ICLR*, 2021. 3