UniDepth: Universal Monocular Metric Depth Estimation Supplementary Material

Luigi Piccinelli[1] Yung-Hsu Yang[1] Christos Sakaridis[1]
Mattia Segu[1] Siyuan Li[1] Luc Van Gool[1,2] Fisher Yu[1]

[1]ETH Zürich [2]INSAIT

This supplementary material offers further insights into our work. In Sec. A we provide results on the official KITTI benchmark, and standard metric evaluation on KITTI and NYU validation set. Moreover, Sec. B includes additional ablations, namely with VIT backbone and a comparison of different pseudo-spherical representations. In addition, differences between convolutional and ViT-based backbones regarding generalization are discussed. In Sec. C, we describe the datasets used for training and testing and how we propose to amend Diode [33] artifacts at boundaries present in ground-truth depth. We analyze the complexity of UniDepth and compare it with other methods in Sec. D. Furthermore, we describe in Sec. E the network architecture in more detail, necessarily Sec. E overlaps with Sec. 3. Eventually, additional visualizations are provided in Sec. F.

A. Results

KITTI benchmark [10]. Table 6 clearly shows the compelling performance of UniDepth on the official KITTI private test set. Results of the latest published methods are reported. The table is fetched from the official KITTI leaderboard for depth prediction. In particular, UniDepth ranks first in the KITTI benchmark at the time of submission among all methods, published and not.

Table 6. **Results on official KITTI [10] Benchmark.** Comparison of performance of methods trained on KITTI and tested on the official KITTI private test set.

Method	SI_{log}	Sq.Rel	A.Rel	iRMS
	Lower is better			
MG [20]	9.93	1.68 %	7.99 %	10.63
URCDC-Depth [28]	10.03	1.74 %	8.24 %	10.71
iDisc [25]	9.89	1.77 %	8.11 %	10.73
VA-DepthNet [21]	9.84	1.66 %	7.96 %	10.44
IEBins [30]	9.63	1.60 %	7.82 %	10.68
NDDepth [29]	9.62	1.59 %	7.75 %	10.62
UniDepth	**8.13**	**1.09%**	**6.54 %**	**8.24**

Table 7. **Comparison on KITTI Eigen-split test set.** The first five methods are trained on KITTI and tested on it. The last six methods are tested in a zero-shot setting. UniDepth-{C, V}: UniDepth-{ConvNext [23], ViT [8]}. (†): MiDaS [26] pre-trained. (‡): predicted intrinsics are utilized for conditioning and backprojecting.

Method	δ_1	δ_2	δ_3	F_A	A.Rel	RMS	RMS_{log}	CD	SI_{log}
	Higher is better				*Lower is better*				
BTS [19]	96.2	99.4	99.8	82.0	5.63	2.43	0.089	0.42	8.18
AdaBins [3]	96.3	99.5	99.8	81.5	5.85	2.38	0.089	0.429	8.10
NeWCRF [40]	97.5	99.7	**99.9**	82.7	5.20	2.07	0.078	0.388	7.00
iDisc [25]	97.5	99.7	**99.9**	83.1	5.09	2.07	0.077	0.380	7.11
ZoeDepth [4]	96.5	99.1	99.4	82.1	5.76	2.39	0.089	0.431	7.47
Metric3D [38]	97.5	99.5	99.8	82.9	5.33	2.26	0.081	0.392	7.28
Ours-C	97.8	99.7	**99.9**	83.9	4.69	2.00	0.073	0.371	6.72
Ours-V	**98.6**	**99.8**	**99.9**	**85.0**	**4.21**	**1.75**	**0.064**	**0.338**	**5.84**
Ours-C ‡	**97.8**	99.7	**99.9**	80.8	4.77	2.00	0.073	0.427	6.72
Ours-V ‡	**98.6**	**99.8**	**99.9**	82.7	**4.21**	**1.75**	**0.064**	0.381	**5.84**

Table 8. **Comparison on NYU validation set.** The first five methods are trained on NYU and tested on it. The last six methods are tested in a zero-shot setting. UniDepth-{C, V}: UniDepth-{ConvNext [23], ViT [8]}. (†): MiDaS [26] pre-trained. (‡): predicted intrinsics are utilized for conditioning and backprojecting.

Method	δ_1	δ_2	δ_3	F_A	A.Rel	RMS	Log_{10}	CD	SI_{log}
	Higher is better				*Lower is better*				
BTS [19]	88.5	**97.8**	99.4	74.0	10.9	0.391	0.046	0.160	11.5
AdaBins [3]	90.1	98.3	99.6	74.7	10.3	0.365	0.044	0.156	10.6
NeWCRF [40]	92.1	99.1	99.8	75.8	9.56	0.333	0.040	0.147	9.16
iDisc [25]	93.8	99.2	99.8	78.2	8.61	0.313	0.037	0.133	8.85
ZoeDepth [4]	95.2	99.5	99.8	80.1	7.70	0.278	0.033	0.125	7.19
Metric3D [38]	92.6	97.9	99.1	77.8	9.38	0.337	0.038	0.146	9.13
Ours-C	97.2	99.6	**99.9**	84.4	6.22	0.231	0.026	0.101	6.39
Ours-V	**98.4**	99.7	**99.9**	**85.9**	**5.78**	**0.201**	**0.024**	**0.092**	**5.27**
Ours-C‡	97.2	99.6	**99.9**	84.1	6.33	0.232	0.027	0.103	6.40
Ours-V‡	98.3	99.7	**99.9**	85.5	6.04	0.205	0.025	0.094	5.28

KITTI Eigen-split and NYUv2-Depth. For the sake of completeness, we report the "standard" metrics results in Table 7 and Table 8 on KITTI Eigen-split and NYU validation set, respectively. It is worth noting that the typical metrics δ_2 and, especially, δ_3 are saturated, thus not informative. Therefore, we advocate our choice of not reporting them in the main paper and prefer to report $\delta_{0.5}$. Moreover, we suggest in future works the use of the area under the curve of the δ metrics as a more informative and comprehensive metric, instead of the values at fixed thresholds, *i.e.* $\{1.25^i\}_{i=1}^3$.

B. Ablations

B.1. Ablations with ViT backbone

Ablations with ViT backbone are provided in Table 9. The trend in Table 9 is consistent with the one outlined for the convolutional backbone. More specifically, the ablated components contribute similarly between ViT-L [8] and ConvNext-L [23] backbones. However, utilizing a ViT backbone shows a larger variability for out-of-domain results, also showing a stronger effect of the usage of pseudo-spherical representation both for the *Baseline* and *Full*. The increased susceptibility of the scene's depth scale to domain shift is also related to the backbone comparison in Table 1. In particular, zero-shot results suggest that the convolutional architecture exhibits superior resilience to scale-related domain shifts, although showing relative disadvantage in handling appearance-related domain shifts. $\mathrm{SI}_{\log}$ consistently favors ViT over convolutional methods, emphasizing the latter's diminished performance in appearance domain shifts. However, scale-dependent metrics do not consistently favor ViT, indicating that the constrained receptive field of convolutional methods yields higher robustness to domain shifts associated with scale.

B.2. Alternative pseudo-spherical representation

Sec. 3 focuses on describing the pseudo-spherical representation chosen to disentangle the two sub-tasks, namely calibration and depth estimation, and ablations studies confirm the effectiveness of disentangling the sub-tasks. In particular, UniDepth exploits an angular pseudo-spherical representation, namely based on azimuth, elevation angle, and log-depth, *i.e.* $(\theta, \phi, z_{\log})$. Nevertheless, an alternative solution to disentangle the two different sub-tasks, namely calibration and depth estimation, is to exploit the bearing vector and log-depth. More specifically, a bearing vector corresponds to the unit-length ray represented by $(r_x, r_y, r_z) \in \mathbb{S}^2$, with $\mathbb{S}^2$ corresponding to the unit-sphere manifold. The bearing vectors are obtained as the unprojection of image coordinates based on the (pinhole) camera model. With this design, the output is represented by the tuple $(r_x, r_y, r_z, z_{\log})$ and the loss $\mathcal{L}_{\lambda\mathrm{MSE}}$ is applied seamlessly as depicted in Sec. 3, but with $\lambda_{r_x} = \lambda_{r_y} = \lambda_{r_z} = 1$ and $\lambda_z = 0.15$.

However, the disentanglement in rays and log-depth can be viewed as an alternative pseudo-spherical representation, in fact, rays and angles share a direct relationship $\theta = \arctan(\frac{r_x}{r_z})$ and $\phi = \arccos(r_y)$. Table 10 explores the effectiveness of this alternative representation and compares to the one presented in Sec. 3. The ablation study reported in Table 10 highlights how the difference between the two representations is marginal and, in most cases, within the uncertainty range, thus proving their similarity. The main difference lies in the output space dimensionality. In principle, the bearing vectors would span the entire $\mathbb{R}^3$ space. However, the space is constrained to the unit-sphere manifold by L_2 normalization.

Furthermore, we ablate our camera prompting with respect to CAMConvs [9] in Table 11

Algorithm 1 GT depth boundaries refining.

procedure BOUNDARYREFINE($\mathbf{Z}_{\log}$)
  $\mathbf{L} = \mathrm{Laplacian}(\mathbf{Z}_{log}, k = 5)$
  $\mathbf{M} = \mathbb{I}[\mathbf{L}_{10\%} \leq \mathbf{L} \leq \mathbf{L}_{90\%}]$ ▷ Compute Laplacian and threshold at 10-90 percentile
  $\mathbf{M} = (\mathbf{M} \ominus \mathrm{eye}_3) \oplus \mathrm{eye}_3$ ▷ Opening with size 3
  $\mathbf{M} = \mathrm{MedianBlur}(\mathbf{M}, k = 3)$
  $\mathbf{Z} = \exp(\mathbf{Z}_{\log}) \cdot \mathbf{M}$
  return Z

C. Datasets

C.1. Datasets details

Details of training and testing datasets are presented in Table 12. The training datasets are processed in a way that the interval between two consecutive RGB and GT depth frames is not smaller than one second. We do not apply any post-processing apart from the aforementioned subsampling. The total amount of training samples accounts for 3'743'000 samples. SUN-RGBD [31] validation set involves also NYU [24] test set. Therefore, we removed the samples corresponding to NYU test set to avoid any overlap between test sets. As per standard practice, KITTI Eigen-split corresponds to the corrected and accumulated GT depth maps with 45 images with inaccurate GT discarded from the original 697 images.

C.2. Diode Indoor ground-truth correction

Diode [33] ground-truth depth is not perfectly accurate on boundaries, in particular, a simple inspection shows how depth in boundaries presents low values, but greater than zero. These artifacts present in the GT affect the validation pipeline and results. Therefore, we design a simple image processing algorithm, outlined in Algorithm 1, that, first, detects the aforementioned boundary artifacts and, second, masks the depth in the corresponding neighborhoods. Thanks to masking those boundaries, the corresponding regions are ignored during validation.

D. Model Complexity

Table 13 displays the parameters and inference complexity of UniDepth and other SotA methods. UniDepth with ViT-L backbone is comparable to ZoeDepth in terms of efficiency and model parameters; however UniDepth surpasses it in terms of performance as stated in Sec. 4. Metric3D displays an improved efficiency due to the fully convolutional and relatively low dimensionality designed in the decoder. It is worth highlighting how ZeroDepth presents a low efficiency

Table 9. **Ablations of UniDepth.** *In-Domain* corresponds to the union of the training domain's validation sets, while *Out-of-Domain* involves the union of zero-shot testing sets. *Oracle* is the model with provided GT cameras at training and test time. *Baseline* directly predicts 3D points in Cartesian space, *Baseline++* in pseudo-spherical. *Full* represents the final UniDepth. All models have the same depth and camera module architecture, if any. ARel_C is the mean of elementwise absolute relative error for camera intrinsics. (†): GT camera intrinsics utilized for backprojection. The backbone used is ViT-L [8]. Medians and median average deviations over three runs are reported.

	Ablation	In-Domain				Out-of-Domain			
		$\delta_1 \uparrow$	$\mathrm{SI}_{\log} \downarrow$	$\mathrm{F}_A \uparrow$	$\mathrm{ARel}_C \downarrow$	$\delta_1 \uparrow$	$\mathrm{SI}_{\log} \downarrow$	$\mathrm{F}_A \uparrow$	$\mathrm{ARel}_C \downarrow$
1	Oracle	91.46±0.09	12.12±0.02	68.35±0.14	n/a	72.17±0.44	13.07±0.01	59.84±0.18	n/a
2	Full	91.43±0.05	12.06±0.06	65.44±0.84	2.19±0.14	64.45±0.52	13.0±0.02	52.46±0.29	12.31±0.61
3	– Camera	89.33±0.04	12.54±0.04	66.02±0.27	n/a	60.67±0.22	13.4±0.07	52.43±0.08	n/a
4	– $\mathcal{L}_{\mathrm{con}}$	90.27±0.13	12.21±0.01	63.28±0.66	1.92±0.31	61.98±0.41	13.24±0.04	50.91±0.16	13.11±0.36
5	– Spherical	32.92±0.18	18.11±0.08	33.62±0.07	21.64±0.2	48.43±1.27	18.53±0.35	42.85±1.18	17.16±0.79
6	– Dense	90.16±0.15	12.23±0.01	64.19±0.03	1.83±0.18	62.44±0.19	13.36±0.04	49.34±0.28	13.68±0.61
7	– Detach	89.93±0.02	12.58±0.04	66.30±0.35	0.94±0.03	51.77±0.09	13.45±0.02	49.91±0.01	14.87±0.22
8	Baseline	21.26±0.23	23.43±0.45	29.19±0.09	n/a	34.15±0.74	20.39±0.42	40.14±0.52	n/a
9	Baseline++	88.84±0.11	12.93±0.11	42.72±0.10	n/a	59.31±0.58	14.04±0.03	44.12±0.10	n/a

Table 10. **Ablations of specific pseudo-spherical representation.** *In-Domain* corresponds to the union of the training domain's validation sets, while *Out-of-Domain* involves the union of zero-shot testing sets. All models have the same depth and camera module architecture. ARel_C is the mean of elementwise absolute relative error for camera intrinsics. Medians and median average deviations over three runs are reported.

Ablation	Backbone	In-Domain				Out-of-Domain			
		$\delta_1 \uparrow$	$\mathrm{SI}_{\log} \downarrow$	$\mathrm{F}_A \uparrow$	$\mathrm{ARel}_C \downarrow$	$\delta_1 \uparrow$	$\mathrm{SI}_{\log} \downarrow$	$\mathrm{F}_A \uparrow$	$\mathrm{ARel}_C \downarrow$
UniDepth	ViT-L [8]	91.43±0.05	12.06±0.06	65.44±0.84	2.19±0.14	64.45±0.52	13.00±0.02	52.46±0.29	12.31±0.61
UniDepth $_{\text{rays}}$	ViT-L [8]	90.93±0.02	12.19±0.06	64.70±0.05	2.44±0.11	65.50±0.81	13.03±0.01	53.12±0.02	11.82±0.99
UniDepth	ConvNext-L [23]	88.89±0.10	13.13±0.01	63.52±0.08	2.05±0.01	57.06±1.48	14.83±0.04	49.71±0.55	13.54±0.85
UniDepth $_{\text{rays}}$	ConvNext-L [23]	88.55±0.31	13.24±0.10	62.58±1.11	2.74±0.13	55.10±0.39	14.91±0.01	46.38±0.61	15.00±0.36

Table 11. **Ablate UniDepth with CAMConvs.** *Full* is complete UniDepth, as row 2 in Tab. 5. *w/ CAMConvs* represents UniDepth with CAMConvs [14] conditioning instead of our prompting.

Ablation	In-Domain				Out-of-Domain			
	$\delta_1 \uparrow$	$\mathrm{SI}_{\log} \downarrow$	$\mathrm{F}_A \uparrow$	$\mathrm{ARel}_C \downarrow$	$\delta_1 \uparrow$	$\mathrm{SI}_{\log} \downarrow$	$\mathrm{F}_A \uparrow$	$\mathrm{ARel}_C \downarrow$
w/ CAMConvs	87.81	13.49	60.90	2.55	54.65	15.37	43.09	16.11
Full	**88.89**	**13.13**	**63.52**	**2.05**	**57.06**	**14.83**	**49.71**	**13.54**

although based on ResNet-18, we argue that this is due to the expensive full-resolution cross-attention in the decoder. The last two rows in Table 13 analyze separately the complexity of the single Camera and Depth Module. The Camera Module is a lightweight component accounting for 13.4M parameters. On the other hand, the Depth Module amounts to more than half of the total latency, despite the limited memory consumption. The Depth Module's high latency is due to the several (6) self-attention layers in the decoder.

E. Network Architecture

Encoder. We show the effectiveness of our method with different encoders, both convolutional and transformer-based ones, *e.g.*, ConvNext [23] and ViT [8]. However, all of them follow the same structure: the feature maps are extracted at each layer and the features map corresponding to a "scale" is obtained as the pixel-wise average. For ConvNext, we obtain the class tokens as the average pooled feature maps. All backbones utilized are originally designed for classification,

Table 12. **Datasets List.** List of the training and testing datasets: number of images, scene type, and method of acquisition are reported. SfM: Structure-from-Motion. MVS: Multi-View Stereo.

	Dataset	Images	Scene	Acquisition
Training Set	A2D2 [11]	78k	Outdoor	LiDAR
	Argoverse2 [34]	403k	Outdoor	LiDAR
	BDD100k [39]	270k	Outdoor	SfM
	CityScapes [6]	24k	Outdoor	MVS
	DrivingStereo [37]	63k	Outdoor	MVS
	Mapillary PSD [1]	742k	Outdoor	SfM
	ScanNet [7]	83k	Indoor	RGB-D
	Taskonomy [41]	1940k	Indoor	RGB-D
	Waymo [32]	223k	Outdoor	LiDAR
Testing Set	DDAD [12]	1002	Outdoor	LiDAR
	Diode [33]	325	Indoor	LiDAR
	ETH3D [27]	454	Outdoor	RGB-D
	HAMMER [17]	496	Indoor	Mix
	IBims-1 [18]	100	Indoor	RGB-D
	KITTI [10]	652	Outdoor	LiDAR
	NuScenes [5]	3k	Outdoor	LiDAR
	NYU [24]	654	Indoor	RGB-D
	SUN-RGBD [31]	4.4k	Indoor	RGB-D
	VOID [35]	800	Indoor	RGB-D

thus we remove the last 3 layers, *i.e.*, the pooling layer, fully connected layer, and $\mathrm{softmax}$ layer. The feature maps are flattened, then LayerNorm [2] (LN) and a linear layer are applied. The linear layer projects the features to a common

Table 13. **Parameters and efficiency comparison.** Comparison of performance of methods based on latency, throughput, and number of trainable parameters. Tested on RTX3090 GPU, 32-bit precision float, and input image with size (480, 640). The last two rows correspond to the Camera and Depth Moudel evaluated independently. R18: ResNet-18 [14], D161: DenseNet-161 [16], EN-B5: EfficientNet-B5-AP [36], CNXT-L: ConvNext-L [23].

Method	Backbone	Latency (ms)	Throughput (FPS)	Parameters (M)
BTS [19]	D161	28.5	35.1	47.0
Adains [3]	EN-B5	33.2	30.1	78.3
NewCRF [40]	SWin-L [22]	53.1	18.8	280.0
iDisc [25]	SWin-L [22]	81.1	12.3	209.2
ZoeDepth [4]	BEiT-L	144.8	6.91	345.9
ZeroDepth [13]	R18	955.6	1.05	232.6
Metric3D [38]	CNXT-L	40.3	24.8	203.2
UniDepth	CNXT-L	86.6	11.5	238.9
UniDepth	ViT-L [8]	146.4	6.83	347.0
Camera Module	-	5.1	-	13.4
Depth Module	-	49.2	-	26.6

channel dimension of 512. The projected feature maps are interpolated to a common shape, namely $(h, w) = (\frac{H}{16}, \frac{W}{16})$, with H, W as input height and width, respectively. Two independent projections are utilized for the features maps, *i.e.* $\mathbf{F} \in \mathbb{R}^{h \times w \times C \times B}$ with B corresponding to the four scales, and C set to 512 as mentioned above, and the class tokens $\in \mathbb{R}^{C \times B}$, the latter fed to the Camera Module only.

Camera Module. The camera parameters are initialized with the four class tokens extracted from the Encoder. The flattened and stacked feature maps from the encoder are detached and used as *keys* and *values* in one cross-attention layer, where the *queries* correspond to the four camera parameters. The output is processed by a MultiLayer Perceptron (MLP) with one hidden layer with dimension of 2048 and non-linear activation Gaussian Error Linear Unit (GELU) [15]. The cross-attention and the MLP present a residual connection. The four tokens are further processed with two additional self-attention layers, projected to dimension one and then exponentiated. The camera parameters are obtained as $f_x = \frac{\Delta f_x W}{2}$, $f_y = \frac{\Delta f_y H}{2}$, $c_x = \frac{\Delta c_x W}{2}$, $c_y = \frac{\Delta c_y H}{2}$. The dense camera representation $\mathbf{C}$ is obtained by backprojecting with the predicted camera parameters: $(\mathbf{r}_x, \mathbf{r}_y, \mathbf{r}_z) = \mathbf{K}^{-1}[\mathbf{u}, \mathbf{v}, \mathbf{1}]^T$ and calculating the azimuth and elevation angles, θ and ϕ, as in Sec. B. The angular representation is embedded through the Laplace Spherical Harmonics Embedding (SHE) leading to 81 channels, resulting in $\mathbf{E} \in \mathbb{R}^{h \times w \times 81}$.

Depth Module. The depth latents are initialized as the average of the features $\mathbf{F}$ along the B dimension. Then, the latents are conditioned on the original feature tensor $\mathbf{F}$ via one cross-attention layer where two projections of $\mathbf{F}$ account for *keys* and *values* and $\mathbf{L}$ as *queries*. In addition, one MLP is applied, seamlessly as in the Camera Module. Furthermore, the depth features are conditioned on the camera prompts $\mathbf{E}$ with one additional cross-attention layer, where *keys* and *values* are two projections of camera embeddings $\mathbf{E}$, and one MLP as above. The features are decoded in three consecutive stages. The first stage applies three self-attention layers with $\mathbf{E}$ as positional encoding. The features are then processed with one ConvNext [23] layer, upsampled by a factor of two, and the channels are halved. The second and third stages are similar, although the second stage presents two self-attention layers and the third only one. In the second and third stages, MLP's hidden channel dimension is sequentially halved, too, from the initial aforementioned value of 2048. Each stage's output is projected to a dimension one. Therefore, the three output maps are interpolated to a common shape, *i.e.* $(\frac{H}{2}, \frac{W}{2})$, and pixel-wise averaged. The final log-depth output $\mathbf{Z}_{\log}$ is obtained by upsampling the obtained tensor to the input shape (H, W). The final depth is element-wise exponentiation of $\mathbf{Z}_{\log}$.

F. Visualization

We provide here twenty more qualitative comparisons, two for each zero-shot test set: KITTI, NYU, Diode, ETH3D in Fig. 5, DDAD, NuScenes, SUN-RGBD, IBims-1 in Fig. 6, and Fig. 7 displays VOID and HAMMER. The error maps are shown after applying median-based rescaling. The rescaling was deemed necessary to avoid some of the error maps being completely red and not informative. Due to sparsity, DDAD and Nuscenes GT and error maps are dilated by a factor of 5, leading to visible GT depth and error maps.

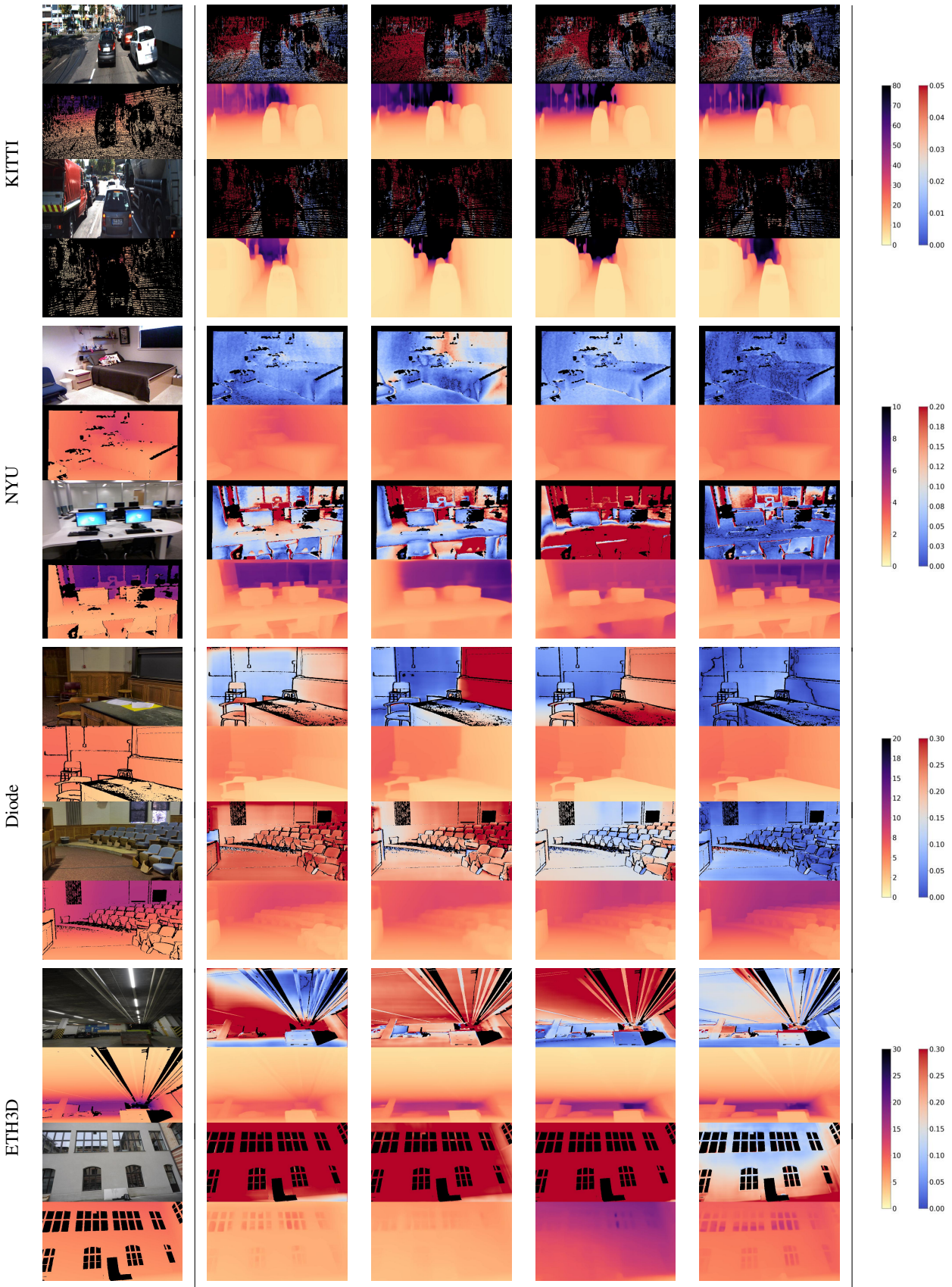

Figure 5. **Zero-shot qualitative results.** Each pair of consecutive rows corresponds to one test sample. Each odd row shows the input RGB image and the absolute relative error map color-coded with *coolwarm* colormap. Each even row shows GT depth and the predicted depth. The last column represents the specific colormap ranges for depth and error. (†): KITTI and NYU in the training set.

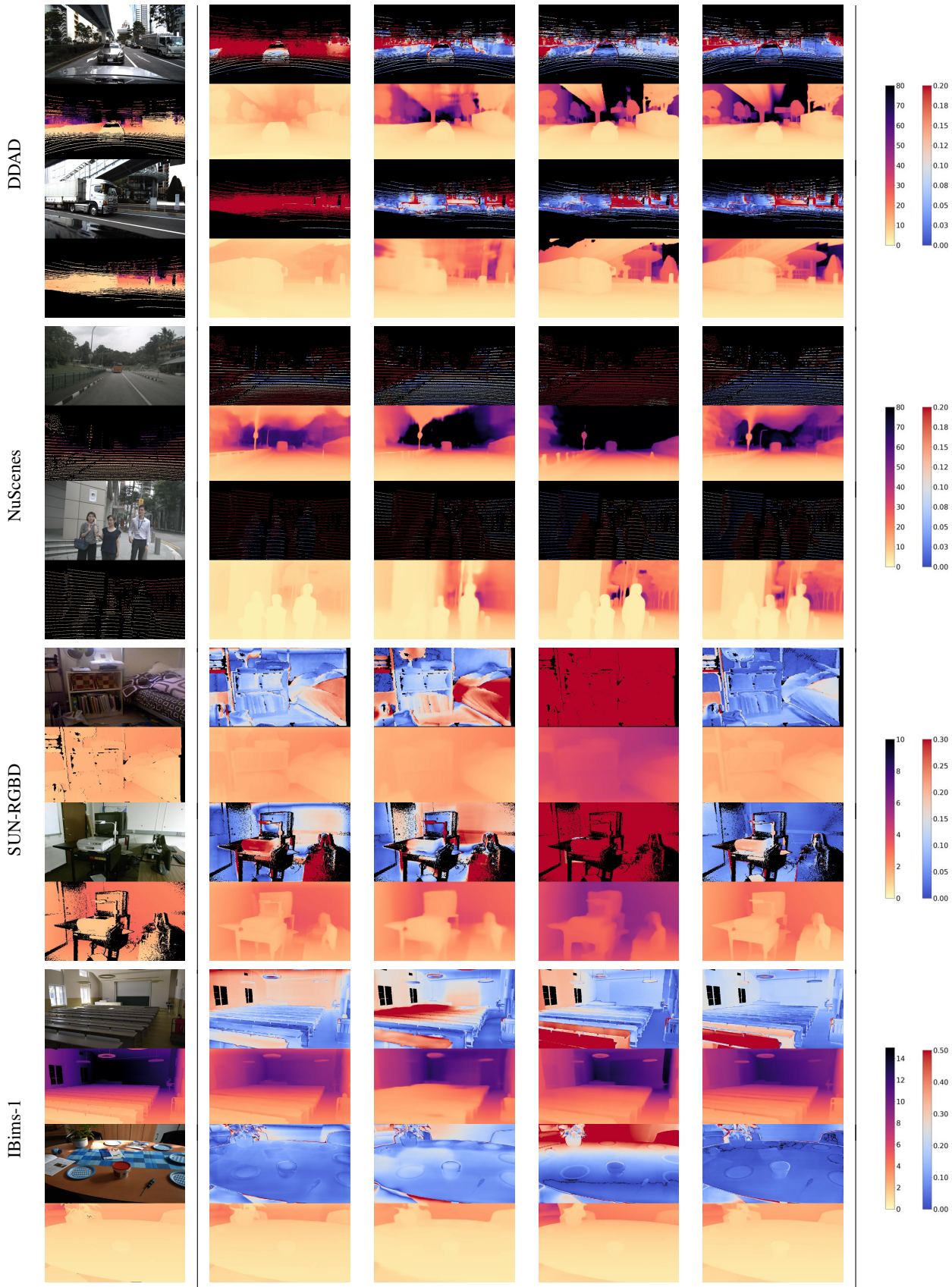

Figure 6. **Zero-shot qualitative results.** Each pair of consecutive rows corresponds to one test sample. Each odd row shows the input RGB image and the absolute relative error map color-coded with *coolwarm* colormap. Each even row shows GT depth and the predicted depth. The last column represents the specific colormap ranges for depth and error. (†): DDAD in the training set.

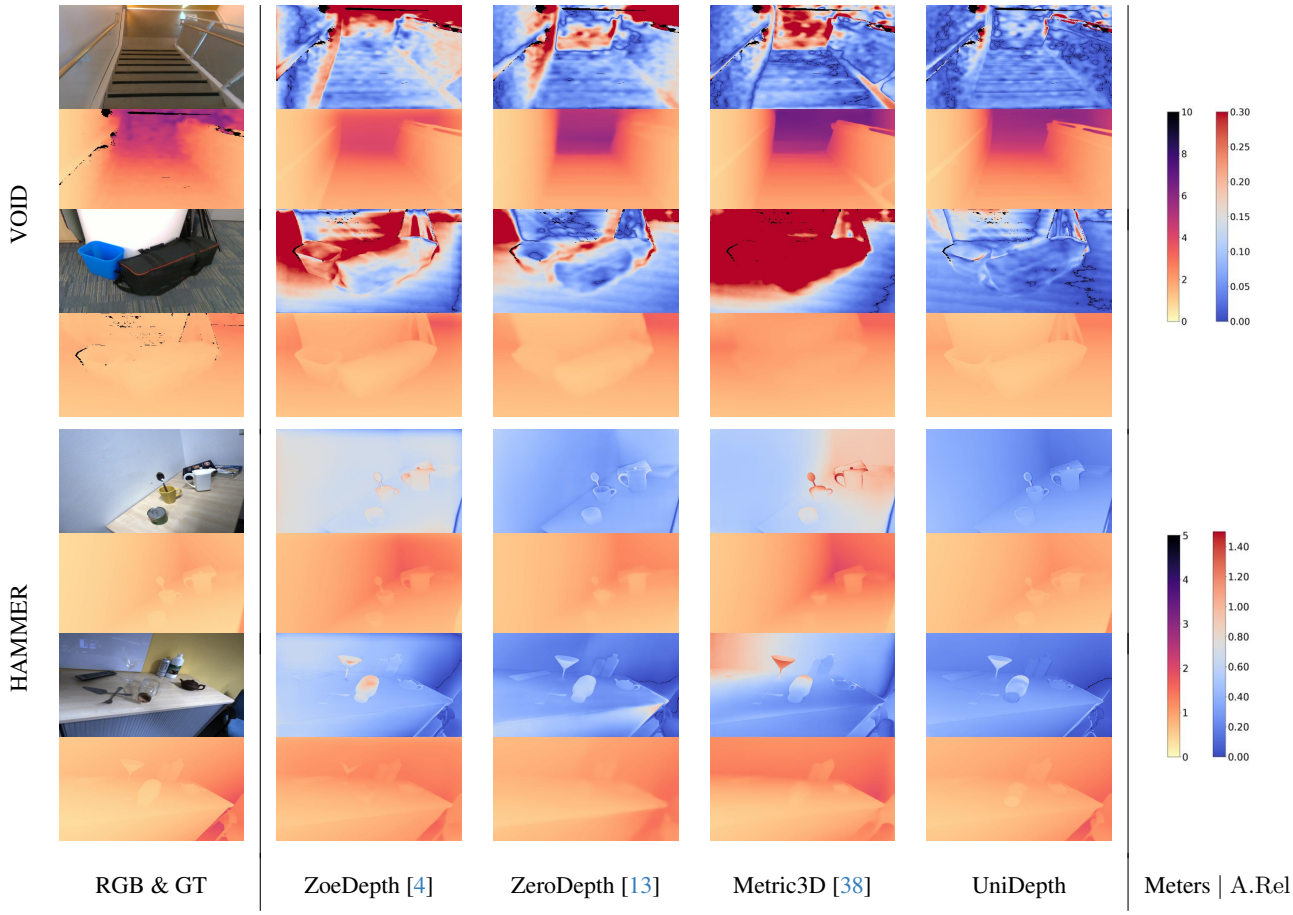

Figure 7. **Zero-shot qualitative results.** Each pair of consecutive rows corresponds to one test sample. Each odd row shows the input RGB image and the absolute relative error map color-coded with *coolwarm* colormap. Each even row shows GT depth and the predicted depth. The last column represents the specific colormap ranges for depth and error.

References

[1] Manuel Lopez Antequera, Pau Gargallo, Markus Hofinger, Samuel Rota Bulò, Yubin Kuang, and Peter Kontschieder. Mapillary planet-scale depth dataset. In *The European Conference Computer Vision (ECCV)*, pages 589–604. Springer International Publishing, 2020. 3

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 3

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

[4] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 1, 4, 5, 6, 7

[5] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3

[7] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2021. 1, 2, 3, 4

[9] Jose M Facil, Benjamin Ummenhofer, Huizhong Zhou, Luis Montesano, Thomas Brox, and Javier Civera. Cam-convs: Camera-aware multi-scale convolutions for single-view depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11826–11835, 2019. 2

[10] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 1, 3

[11] Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S. Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, Tiffany Fernandez, Martin Jänicke, Sudesh Mirashi, Chiragkumar Savani, Martin Sturm, Oleksandr Vorobiov, Martin Oelker, Sebastian Garreis, and Peter Schuberth. A2D2: Audi Autonomous Driving Dataset. *arXiv preprint arXiv:2004.06320*, 2020. 3

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

[13] Vitor Guizilini, Igor Vasiljevic, Dian Chen, Rareș Ambruș, and Adrien Gaidon. Towards zero-shot scale-aware monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9233–9243, 2023. 4, 5, 6, 7

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016-December:770–778, 2015. 4

[15] Dan Hendrycks and Kevin Gimpel. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *CoRR*, abs/1606.08415, 2016. 4

[16] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. *IEEE Conference on Computer Vision and Pattern Recognition*, 2017-January:2261–2269, 2016. 4

[17] HyunJun Jung, Patrick Ruhkamp, Guangyao Zhai, Nikolas Brasch, Yitong Li, Yannick Verdie, Jifei Song, Yiren Zhou, Anil Armagan, Slobodan Ilic, et al. Is my depth ground-truth good enough? HAMMER – Highly Accurate Multi-Modal dataset for dEnse 3D scene Regression. *arXiv preprint arXiv:2205.04565*, 2022. 3

[18] Tobias Koch, Lukas Liebel, Marco Körner, and Friedrich Fraundorfer. Comparison of monocular depth estimation methods using geometrically relevant metrics on the IBims-1 dataset. *Computer Vision and Image Understanding (CVIU)*, 191:102877, 2020. 3

[19] 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. *CoRR*, abs/1907.10326, 2019. 1, 4

[20] Ce Liu, Suryansh Kumar, Shuhang Gu, Radu Timofte, and Luc Van Gool. Single image depth prediction made better: A multivariate gaussian take. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17346–17356, 2023. 1

[21] Ce Liu, Suryansh Kumar, Shuhang Gu, Radu Timofte, and Luc Van Gool. Va-depthnet: A variational approach to single image depth prediction. *arXiv preprint arXiv:2302.06556*, 2023. 1

[22] 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/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021. 4

[23] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11976–11986, 2022. 1, 2, 3, 4

[24] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *The European Conference Computer Vision (ECCV)*, 2012. 2, 3

[25] Luigi Piccinelli, Christos Sakaridis, and Fisher Yu. iDisc: Internal discretization for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 4

[26] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 44(3):1623–1637, 2020. 1

[27] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[28] Shuwei Shao, Zhongcai Pei, Weihai Chen, Ran Li, Zhong Liu, and Zhengguo Li. Urcdc-depth: Uncertainty rectified cross-distillation with cutflip for monocular depth estimation. *arXiv preprint arXiv:2302.08149*, 2023. 1

[29] Shuwei Shao, Zhongcai Pei, Weihai Chen, Xingming Wu, and Zhengguo Li. Nddepth: Normal-distance assisted monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7931–7940, 2023. 1

[30] Shuwei Shao, Zhongcai Pei, Xingming Wu, Zhong Liu, Weihai Chen, and Zhengguo Li. Iebins: Iterative elastic bins for monocular depth estimation. *arXiv preprint arXiv:2309.14137*, 2023. 1

[31] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 07-12-June-2015:567–576, 2015. 2, 3

[32] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2446–2454, 2020. 3

[33] Igor Vasiljevic, Nicholas I. Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. DIODE: A dense indoor and outdoor depth dataset. *CoRR*, abs/1908.00463, 2019. 1, 2, 3

[34] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Advances in Neural Information Processing Systems*, 2021. 3

[35] Alex Wong, Xiaohan Fei, Stephanie Tsuei, and Stefano Soatto. Unsupervised depth completion from visual inertial odometry. *IEEE Robotics and Automation Letters (RA-L)*, 5(2):1899–1906, 2020. 3

[36] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L. Yuille, and Quoc V. Le. Adversarial examples improve image recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 816–825, 2019. 4

[37] Guorun Yang, Xiao Song, Chaoqin Huang, Zhidong Deng, Jianping Shi, and Bolei Zhou. Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[38] Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9043–9053, 2023. 1, 4, 5, 6, 7

[39] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2636–2645, 2020. 3

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

[41] Amir R Zamir, Alexander Sax, William B Shen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018. 3