Robust Depth Enhancement via Polarization Prompt Fusion Tuning

Supplementary Materials

Kei Ikemura[1] Yiming Huang[2] Felix Heide[3]
Zhaoxiang Zhang[4,5] Qifeng Chen[6] Chenyang Lei[3,5]

[1]KTH Royal Institute of Technology [2]CUHK [3]Princeton University
[4]CAIR, HKISI-CAS [5]CASIA [6]HKUST

A. Contents

We organize this supplementary document as follows:

- Section B provides background on Shape-from-Polarization (SfP).
- Section C includes the details of our training implementation.
- Section D explains additional details of our architecture.
- Section E studies our Polarization Prompt Fusion Tuning (PPFT) as a parameter-efficient fine-tuning method.
- Section F presents results on unseen synthetic data derived from the HAMMER dataset [7].
- Section G presents additional qualitative results from our ablation studies.
- Section H provides discussion on alternative dataset to evaluate this work.

B. Shape from Polarization

B.1. Polarization States

The raw polarization measurement $\mathbf{I}_{\text{pol}}$ is converted to an intensity image $\mathbf{I}_{\text{un}}$, Angle of Linear Polarization (AoLP) (ϕ), and Degree of Linear Polarization (DoLP) (ρ) using Equation 1, 2, and 3 respectively [1].

$$\mathbf{I}_{\text{un}} = \frac{\mathbf{I}_0 + \mathbf{I}_{\frac{\pi}{4}} + \mathbf{I}_{\frac{\pi}{2}} + \mathbf{I}_{\frac{3\pi}{4}}}{2}, \tag{1}$$

$$\phi = \frac{1}{2}\arctan\left(\frac{\mathbf{I}_{\frac{\pi}{4}} - \mathbf{I}_{\frac{3\pi}{4}}}{\mathbf{I}_0 - \mathbf{I}_{\frac{\pi}{2}}}\right), \tag{2}$$

$$\rho = \frac{\sqrt{(\mathbf{I}_0 - \mathbf{I}_{\frac{\pi}{2}})^2 + (\mathbf{I}_{\frac{\pi}{4}} - \mathbf{I}_{\frac{3\pi}{4}})^2}}{\mathbf{I}_{\text{un}}}. \tag{3}$$

B.2. Importance of the Viewing Direction

The viewing direction, $\mathbf{V}$, is a 3-channel map, where the value at each pixel location (u, v) is a 3-element vector representing the light incident direction. This can be computed from the 3-by-3 camera intrinsic matrix, K, as shown in Equation 4, where $\mathbf{H}_{(u,v)}$ is a vector of the pixel location in the homogeneous coordinate, i.e., $(u, v, 1)$.

$$\mathbf{V}_{(u,v)} = K^{-1}\mathbf{H}_{(u,v)} \tag{4}$$

Assuming perspective projection, the viewing direction alone allows us to determine the surface normal direction from polarization data. This is because both the angle of linear polarization (ϕ) and the degree of linear polarization (ρ) are functions of surface normal as well as viewing direction.

Degree of Linear Polarization. The DoLP is a function of the refractive index η and the viewing angle θ^v, see Equation 5 and 6 for diffuse and specular dominant reflection cases respectively [1].

$$\rho = \frac{(\eta - \frac{1}{\eta})^2 \text{sin}^2\theta^v_{(u,v)}}{2 + 2\eta^2 - (\eta + \frac{1}{\eta})^2 \text{sin}^2\theta^v_{(u,v)} + 4\text{cos}\theta^v_{(u,v)} C} \tag{5}$$

$$\rho = \frac{2\text{sin}^2\theta^v_{(u,v)}\text{cos}\theta^v_{(u,v)} C}{\eta^2 - \text{sin}^2\theta^v_{(u,v)} - \eta^2\text{sin}^2\theta^v_{(u,v)} + 2\text{sin}^4\theta^v_{(u,v)}} \tag{6}$$

where

$$C = \sqrt{\eta^2 - \text{sin}^2\theta^v_{(u,v)}}. \tag{7}$$

The viewing angle, θ^v, is dependent on the viewing direction as well as the surface normal, as in Equation 8.

$$\begin{aligned} \text{cos}\theta^v_{(u,v)} &= \mathbf{n}_{(u,v)} \cdot \mathbf{V}_{(u,v)} \\ &= \mathbf{n}^x_{(u,v)}\mathbf{V}^x_{(u,v)} + \mathbf{n}^y_{(u,v)}\mathbf{V}^y_{(u,v)} \\ &+ \mathbf{n}^z_{(u,v)}\mathbf{V}^z_{(u,v)} \end{aligned} \tag{8}$$

Angle of Linear Polarization. Following [8], the AoLP, ϕ can be computed via Equation 9, where $\mathbf{\Phi}$ is calculated based on Equation 10, where $\mathbf{n}^c$ is the surface normal of the image plane, i.e., $(0, 0, 1)$. We can observe that in either specular-dominant or diffuse dominant reflections, the AoLP is dependent on the viewing direction.

$$\phi_{(u,v)} = \text{arctan}(\mathbf{\Phi}^y_{(u,v)}, \mathbf{\Phi}^x_{(u,v)}) \tag{9}$$

$$\mathbf{\Phi}_{(u,v)} = \begin{cases} \mathbf{n}_{(u,v)} \times \mathbf{V}_{(u,v)} \times \mathbf{n}^c, & \text{diffuse} \\ \mathbf{n}_{(u,v)} \times \mathbf{V}_{(u,v)} \times \mathbf{V}_{(u,v)} \times \mathbf{n}^c, & \text{specular} \end{cases} \tag{10}$$

C. Training Details

C.1. HAMMER Dataset

Using the HAMMER dataset [7], we use 4× down-sampled images for all inputs, that is, from (832, 1088) to (272, 208) by slicing the input spatially with step 4. We train 250 epochs for all methods. A batch size of 14 per GPU is used on two NVIDIA 4090 GPUs. We use AdamW optimizer [10] with initial learning rate of 10^{-3} and weight decay of 10^{-2}. In addition, we adopted the cosine scheduler [11] to decay our learning rate.

C.2. SPW Dataset

We evaluate our proposed strategy on SfP tasks, using the SPW dataset [8]. For this, we train our model for 8000 iterations. We use a single NVIDIA A40 GPU with a batch size of 20 for training. For both training and test sets, the original frames of size 1224 × 1024 are used. We use a different data split than adopted by the original work [8], which leads to different quantitative results as presented in [8]. Note that nevertheless, the split is consistent across all models, including on the reported SfP-Wild. Also, please note that our focus is on the gain by adopting our proposed PPFT.

D. Details of Network Architecture

We provide the detailed architecture and network parameters of our proposed method in Table 1). We apply the same 3 × 3 convolutional layer as the RGB and depth input for the input polarization embedding. In order to align the input between each encoder layer, we apply a 3 × 3 convolutional layer with a stride of 2, which has the same output channel dimension as the output from the original encoder layer. While the foundation backbone is kept as the same design of CompletionFormer [17], the proposed method can be directly applied in other ViT [2] based backbones to enable cross-modality transfer learning.

Name	Backbone Layers	Fusion Layers	Output dimension
RGB and Depth Embedding			
input	—	—	Polarization: $H \times W \times 7$ RGB Image: $H \times W \times 3$ Sensor Depth: $H \times W \times 1$
conv_separate	RGB: Conv $3 \times 3, 48$ Sensor Depth: Conv $3 \times 3, 16$	Polarization: Conv $3 \times 3, 48$	Polarization Feature: $H \times W \times 48$ RGB Feature: $H \times W \times 48$ Depth Feature: $H \times W \times 16$
conv1	concat [RGB + Polarization, Depth Feature] Conv$3 \times 3, 64$	—	$H \times W \times 64$
Encoder			
conv2	ResNet34 [4] Block $\times 3$	—	$H \times W \times 64$
conv3	ResNet34 [4] Block $\times$ 4 PPFB (RGB, Polarization Feature)	Conv 3×3, stride $= 2,\ 128$ PPFB Block	$\frac{1}{2}H \times \frac{1}{2}W \times 128$
conv4	IJCAT Blocks [17] PPFB (RGB, Polarization Feature)	Conv 3×3, stride $= 2,\ 64$ PPFB Block	$\frac{1}{4}H \times \frac{1}{4}W \times 64$
conv5	IJCAT Blocks [17] PPFB (RGB, Polarization Feature)	Conv 3×3, stride $= 2,\ 128$ PPFB Block	$\frac{1}{8}H \times \frac{1}{8}W \times 128$
conv6	IJCAT Blocks [17] PPFB (RGB, Polarization Feature)	Conv 3×3, stride $= 2,\ 320$ PPFB Block	$\frac{1}{16}H \times \frac{1}{16}W \times 320$
conv7	IJCAT Blocks [17] PPFB (RGB, Polarization Feature)	Conv 3×3, stride $= 2,\ 512$ PPFB Block	$\frac{1}{32}H \times \frac{1}{32}W \times 512$
Decoder			
dec6	ConvTranspose 3×3, stride $= 2,\ 256$ Convolutional Attention Layer	—	$\frac{1}{16}H \times \frac{1}{16}W \times 256$
dec5	concat [dec6, conv6] ConvTranspose 3×3, stride $= 2,\ 128$ Convolutional Attention Layer	—	$\frac{1}{8}H \times \frac{1}{8}W \times 128$
dec4	concat [dec5, conv5] ConvTranspose 3×3, stride $= 2,\ 64$ Convolutional Attention Layer	—	$\frac{1}{4}H \times \frac{1}{4}W \times 64$
dec3	concat [dec4, conv4] ConvTranspose 3×3, stride $= 2,\ 64$ Convolutional Attention Layer	—	$\frac{1}{2}H \times \frac{1}{2}W \times 64$
dec2	concat [dec3, conv3] ConvTranspose 3×3, stride $= 2,\ 64$ Convolutional Attention Layer	—	$H \times W \times 64$
Initial Depth, Confidence, Non-local Neighbors, Affinity Prediction Head			
dec1	concat [dec2, conv2] Conv $3 \times 3, 64$	—	$H \times W \times 64$
dec0	concat [dec1, conv1] Conv $3 \times 3, \eta$	—	$H \times W \times \eta$
SPN Refinement			
refine	NLSPN [13] with recurrent time $K = 6$	—	$H \times W \times 1$

Table 1. **Network Architecture Details of Proposed Polarization Prompt Fusion Tuning (PPFT) Method.** Here, 'concat' denotes the concatenate operation along the channel dimension.

E. Parameter Efficient Fine-Tuning

We also investigate our proposed approach as a Parameter-Efficient Fine-Tuning (PEFT) [3, 16] method. Specifically, we freeze all pre-trained foundation weights during training and only update the parameters in our proposed Polarization Prompt Fusion Block (PPFB) as well as the polarization embedding layers. We compare this against two parameter-efficient fine-tuning methods based on Visual Prompt Tuning (VPT) methods, namely VPT [6] and ViPT [18]. As Table 2 reports, the proposed cross-modal transfer learning module achieves a favorable response compared to training the model on the HAMMER [7] dataset from scratch. The freezed version of our proposed method obtains a more significant performance gain at the cost of only a small increase in the number of trainable parameters compared to VPT [6] and ViPT [18]. We observe that methods based on Visual Prompt Tuning (VPT) [6, 18] are not able to fit complex dense geometry prediction problems which agree with the observation by Jia *et al.* [6]. Qualitative results of various PEFT methods are presented in Figure 1.

Method	RMSE	δRMSE	MAE	δMAE	Parameters
		(mm)↓			
[17]*	313.60	-	241.40	-	-
[17]†	41.17	-	22.14	-	82.40M (100%)
VPT [6]	63.78	+22.61	38.12	+15.98	6.35M (7.7%)
ViPT [18]	52.30	+11.13	30.39	+8.25	**2.15M (2.6%)**
Ours‡	**40.71**	**-0.46**	**22.07**	**-0.07**	7.32M (8.9%)

Table 2. **Ablation Experiments for Cross-modal Transfer Learning.** Parameters denote the number of trainable parameters, with the number in parenthesis indicating the proportion of the trainable parameters w.r.t. to all parameters. With our proposed Polarization Prompt Fusion Tuning (PPFT) the performance is improved significantly. * denotes that the foundation is pre-trained on the NYU-Depth V2 dataset [12]. † denotes that the model is trained from scratch with RGB images from the HAMMER dataset [7], ‡ denotes that the foundation is freezed.

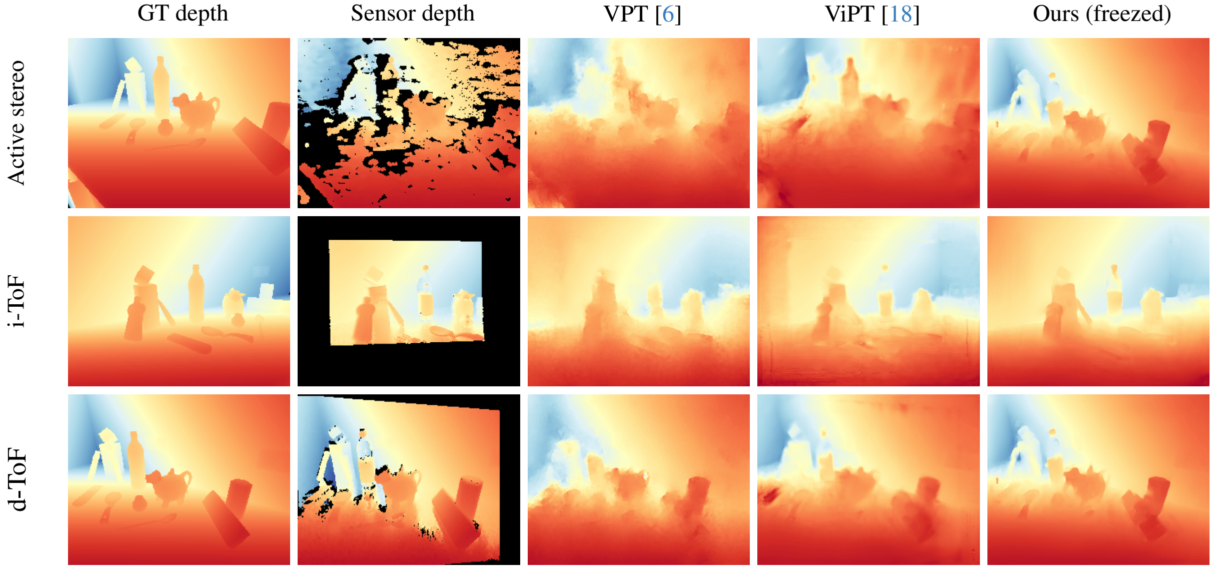

Figure 1. **Evaluation of Proposed Method as a Parameter-efficient Fine-tuning Method.** Our proposed method, with the pre-trained foundation model frozen during the training process, shows competitive performance on the dense geometry prediction problem studied.

Scanning Lines	Model	RMSE (mm)↓	MAE (mm)↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
8	DySPN [9]	225.00	166.13	0.518	0.776	0.909
	CompletionFormer [17]	109.60	78.94	0.855	0.990	0.995
	CompletionFormer† [17]	147.40	106.40	0.723	0.947	0.999
	Ours	**60.04**	**38.38**	**0.968**	**0.998**	**0.999**
16	DySPN [9]	238.52	179.50	0.466	0.735	0.883
	CompletionFormer [17]	93.00	67.56	0.907	0.995	0.999
	CompletionFormer† [17]	147.30	107.40	0.711	0.953	0.999
	Ours	**55.07**	**39.13**	**0.980**	**0.999**	**1.000**

Table 3. **Quantitative Comparison on Synthetic LiDAR Depth Derived from the HAMMER [7] Dataset.** To compare, we select the best three models [9, 17] from the previous tested backbones. Our method obtains the best performance in both synthetic cases. † denotes the model is fine-tuned with RGB images.

F. Additional Testing on Unseen Synthetic Patterns

Inspired by [5], we generate two types of perturbations to the ground truth depth map to simulate different sparse depth measurements by LiDAR. Specifically, we keep 8 and 16 scan lines of depth and remove all of the other depth measurements to simulate the measurement sparsity of typical LiDAR sensors. We directly test all models trained with the HAMMER dataset

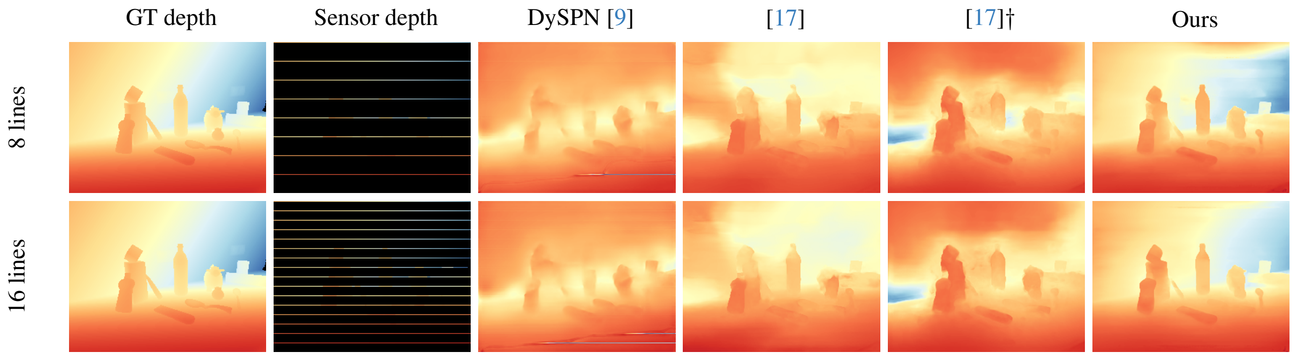

Figure 2. **Evaluation of Proposed Method on Unseen Synthetic Data.** We simulate sparse measurement from typical LiDAR sensors by only keeping 8 and 16 scan lines of measurement. Our proposed method shows superior generalization ability to unseen depth degradation patterns. † denotes that the model has been fine-tuned with pre-trained weights on a large-scale RGB-D dataset.

[7] on the synthetic data. Quantitative results on the synthetic dataset are presented in Table 3. Our method consistently obtains better performance than the three best methods from the previous real sensor depth comparison, where the highest performance (38.38 and 39.13 on the MAE metric) has been achieved on 8, 16 lines LiDAR patterns, respectively. This shows that our method generalizes to complete depth measurement, even on the patterns that are not occurring in the training. Moreover, our polarization-guided depth completion method achieves the best for LiDAR patterns on inputs with different sparsity levels, suggesting that our Polarization Prompt Fusion Tuning (PPFT) strategy offers a robust solution across varying sparsities.

G. Additional Qualitative Results

In Figure 3, we present additional qualitative results of using the shallow version of our proposed PPFT against the full pipeline (i.e., the deep version). With the shallow version, we observe weaker generalization capability under irregularities (e.g., transparency) and detailed regions, for example, the fork highlighted in Row 2. However, the shallow version of the proposed method still performs adequately.

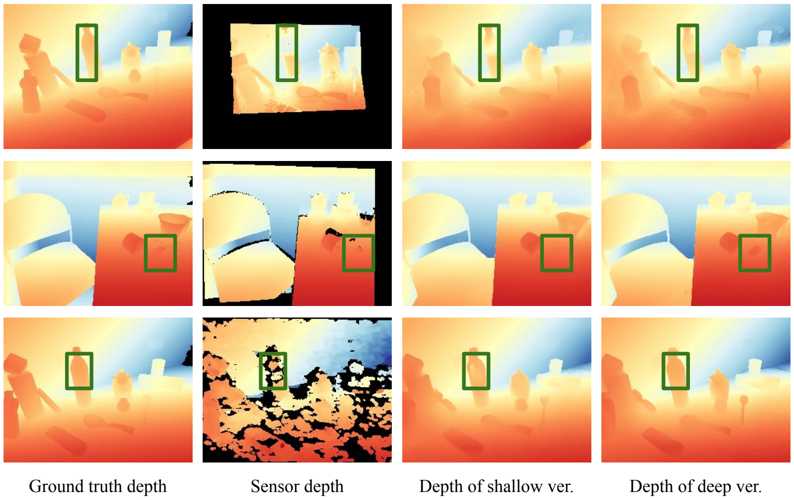

Figure 3. **Qualitative Results of the Shallow and Deep Version of PPFT.** Green boxes highlight the regions to emphasize. Comparisons of the deep and shallow versions of our method are presented. The shallow version shows poorer generalizability towards irregularities (e.g. transparency) and detailed structures.

H. Discussion on Additional Dataset

In the course of this work, we have also considered two additional datasets that provide depth and polarization measurement, namely CroMo [15] and DPS-Net [14]. However, they have been excluded from this study after consideration, this is based on the following argument: Our method can be applied to datasets with paired low-quality sensor depths and high-quality GT depths. However, CroMo on one hand lacks high-quality ground truth depths on challenging areas (e.g., on transparent objects); while DPS-Net, despite having good quality ground truth depth, does not provide paired low-quality sensor depths.

References

[1] Yunhao Ba, Alex Gilbert, Franklin Wang, Jinfa Yang, Rui Chen, Yiqin Wang, Lei Yan, Boxin Shi, and Achuta Kadambi. Deep shape from polarization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 554–571. Springer, 2020. 1, 2

[2] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[3] Haoyu He, Jianfei Cai, Jing Zhang, Dacheng Tao, and Bohan Zhuang. Sensitivity-aware visual parameter-efficient fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11825–11835, 2023. 3

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 3

[5] Saif Imran, Xiaoming Liu, and Daniel Morris. Depth completion with twin surface extrapolation at occlusion boundaries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2583–2592, 2021. 4

[6] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022. 3, 4

[7] HyunJun Jung, Patrick Ruhkamp, Guangyao Zhai, Nikolas Brasch, Yitong Li, Yannick Verdie, Jifei Song, Yiren Zhou, Anil Armagan, Slobodan Ilic, Ales Leonardis, and Benjamin Busam. Is my depth ground-truth good enough? hammer – highly accurate multi-modal dataset for dense 3d scene regression, 2022. 1, 2, 3, 4, 5

[8] Chenyang Lei, Chenyang Qi, Jiaxin Xie, Na Fan, Vladlen Koltun, and Qifeng Chen. Shape from polarization for complex scenes in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12632–12641, 2022. 2

[9] Yuankai Lin, Tao Cheng, Qi Zhong, Wending Zhou, and Hua Yang. Dynamic spatial propagation network for depth completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1638–1646, 2022. 4, 5

[10] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2

[11] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts, 2017. 2

[12] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 4

[13] Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kuei Liu, and In So Kweon. Non-local spatial propagation network for depth completion. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 120–136. Springer, 2020. 3

[14] Chaoran Tian, Weihong Pan, Zimo Wang, Mao Mao, Guofeng Zhang, Hujun Bao, Ping Tan, and Zhaopeng Cui. Dps-net: Deep polarimetric stereo depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3569–3579, 2023. 6

[15] Yannick Verdié, Jifei Song, Barnabé Mas, Benjamin Busam, Aleš Leonardis, and Steven McDonagh. Cromo: Cross-modal learning for monocular depth estimation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3927–3937. IEEE, 2022. 6

[16] Dongshuo Yin, Xueting Han, Bin Li, Hao Feng, and Jing Bai. Parameter-efficient is not sufficient: Exploring parameter, memory, and time efficient adapter tuning for dense predictions. *arXiv preprint arXiv:2306.09729*, 2023. 3

[17] Youmin Zhang, Xianda Guo, Matteo Poggi, Zheng Zhu, Guan Huang, and Stefano Mattoccia. Completionformer: Depth completion with convolutions and vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18527–18536, 2023. 2, 3, 4, 5

[18] Jiawen Zhu, Simiao Lai, Xin Chen, Dong Wang, and Huchuan Lu. Visual prompt multi-modal tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9516–9526, 2023. 3, 4