RIAV-MVS: Recurrent-Indexing an Asymmetric Volume for Multi-View Stereo - Supplementary Material -

Changjiang Cai, Pan Ji, Qingan Yan, Yi Xu
OPPO US Research Center, InnoPeak Technology, Inc.

In this supplementary material, we show more details about datasets, network architectures and hyperparameters, ablation studies, and additional qualitative results.

1. Datasets

Our experiments use four indoor-scene datasets, which have RGB-D video frames with ground truth depths and known camera poses. ScanNet [3] and DTU [8] are used in training and testing, and 7scenes [7] and RGB-D Scenes V2 [10] are evaluated for zero-shot generalization.

ScanNet. Our network is trained from scratch on ScanNet [3] using the official training split. Following the frame selection heuristic in [4], considering appropriate view frustum overlap and sufficient baselines, we sample 279,494 training samples and 20,000 validation ones. Each sample contains 3 frames, with one as a reference frame and the others as source frames. For testing, we use ScanNet's official test split (with 100 sequences from scene707 to scene806) and sample every 10 frames following [12], resulting in 20,668 samples for quantitative evaluation. ScanNet has images in 640×480 resolution. In training, they are resized to 256×256 with cropping following [4]. For inference, the input images are resized to 320×256 without cropping. The predicted depth maps are upsampled with nearest neighbor interpolation to the original resolution 640×480 before calculating the quantitative metrics.

DTU. DTU [8] is a smaller dataset compared with ScanNet, but with accurate ground truth depth and pose obtained by a structured light scanner. Following [16–18], the depth range for sampling depth hypotheses is set to $d_{min} = 0.425$ and $d_{max} = 0.935$ meters. Based on the view selection and robust training strategy in [15, 16], we sample 27,097 training samples, 6,174 validation ones, and 1,078 ones for evaluation. Each sample has 5 frames. Input image size is 512×256 in network training, and 640×512 for inference and upsampled with nearest neighbor interpolation to the original size 1600× 1152 for evaluation. To coordinate with ScanNet [3] and 7scenes [7], we use the same depth evaluation metrics proposed in [5].

7-Scenes. We select 13 sequences from 7-Scenes for zero-shot generalization. The valid depth range is set the same as that on ScanNet. We generate a test set with 1,610 samples (each with 5 frames, at 640×480 resolution) by sampling the sequences every 10 frames.

RGB-D Scenes V2. It contains indoor scenes, including chair, sofa, table, bowls, caps, cereal boxes, coffee mugs, and soda cans, etc. We select 8 sequences for testing. Similarly, we sample the video sequence every 10 frames to generate 610 testing samples (each with 5 frames).

2. Experimental Setup

Implementation Details: Our model is implemented using PyTorch [13], and trained end-to-end with a mini-batch size of 8 per NVIDIA RTX A6000 GPU. During training, we use the AdamW optimizer and clip gradients to the range of $[-1, 1]$. When generating the cost volume by plane-sweep stereo, we set the plane hypotheses number as M_0=64. When predicting the final depth using the index field, we set the plane hypotheses number as M_1=256. The same hyperparameters as in [11] are adopted for the context network and 3-level GRU architecture.

Training Schedule: Our network is trained for 20 epochs, with an initial learning rate of 1e-4 and decayed by half at epoch 4^{th} and 8^{th}, respectively. For a fair comparison, we also train the baselines PairNet [4] and IterMVS [15] on the same training samples of ScanNet for 20 epochs, using the official codes. For the baseline PairNet we follow the suggested learning rate scheduler, and for the baseline IterMVS, we use a learning rate of 1e-4, which is decayed by half at epoch 4^{th} and 8^{th}.

3. Our Modules Improve Existing Backbones

Our proposed residual pose module and asymmetric attention module can help improve existing state-of-the-art methods. Here we take two baselines - IterMVS [15] and MVSNet [17] as the backbone. Tab. 1-(a) shows the improved accuracy on the ScanNet test set [3] due to incorporating our residual pose module (*i.e.*, *+pose*) and our asymmetric attention module (*i.e.*, +*atten*). Results in parenthesis and highlighted by gray, denote the residual pose is only used for network training but not for inference [1]. Note that they are listed for reference only, and are not used for comparison with the numbers on other rows. We can see our +*pose* and +*atten* can always boost the baseline backbones on the ScanNet test set. Tab. 1-(b) shows the evaluation on DTU test set [8]. Our +*atten* always helps improve the baselines. Our +*pose* can boost the baseline IterMVS [15], but achieves no obvious improvement on baseline MVSNet [17], probably because the ground truth poses are accurate enough, and the features are concatenated when constructing the cost volume, which is different from the dot production of features in ours and baseline IterMVS.

4. Network Architectures

Multi-scale Feature Fusion Layer. The fusion layer $\mathcal{G}$ aggregates multi-scale features $f_{i,2} \in \mathbb{R}^{H/2 \times W/2 \times F_0}$, $f_{i,4} \in \mathbb{R}^{H/4 \times W/4 \times F_0}$, $f_{i,8} \in \mathbb{R}^{H/8 \times W/8 \times F_0}$ and $f_{i,16} \in \mathbb{R}^{H/16 \times W/16 \times F_0}$ into a matching feature $f_i \in \mathbb{R}^{H/4 \times W/4 \times F_1}$ at 1/4 scale. Here F_0=32 and F_1=128 for feature channels, $i = 0$ for the reference image, and $i = 1, \dots, N-1$ for the source images. The architecture is shown in Fig. 1, including up- and down-sampling, concatenation along the feature channel, a convolution layer Conv0 (with kernel size 3×3, in- and out- channels 128/128), batch normalization, ReLU, and another convolution layer Conv1 (with kernel size 1×1, in- and out- channels 128/128).

Context Feature Network C-Net. We use the context feature network as in [9, 11, 14], which consists of several residual blocks. It contains around 4.32M parameters.

Model Capacity. As shown in Tab. 2, the total number of parameters in our network is 27.6M, where residual pose network takes up 47.18%, GRU-based optimizer takes up 25.20%, and the transformer block takes up 1.25%. If not considering the residual pose net, our model then has 14.57M parameters, and most of them are assigned to GRU-based updater, and fewer capacities are on feature extractors. This kind of capacity configuration makes our model not specialized to one domain (for feature extraction), and is well generalized to unseen domains due to the learning to optimize anchored at cost volume via the GRU-based optimizer to predict the index fields for iteratively improved matching.

Network Training and Log Summary. Our network is trained from scratch on the ScanNet training set (with 279,494 samples). It takes around 2 days on 4 NVIDIA RTX A6000 GPUs for up to 20 epochs of training. The GRU iteration number is set to 12 for training. The total batch size is 32 (*i.e.*, 8 per GPU). Training image size is 256×256. We show the log summary of network training at the last logging step (i.e., step=99,609). From the top to bottom, Fig. 2 shows a batch of input samples (batch size = 4 for logging), including reference images I_0 and two source images I_1 and I_2, the ground truth depth maps and our depth predictions. The residual pose net is supervised by the photometric loss as shown in Fig. 3. We do one epoch of warmup training only for the residual pose net with other layers frozen.

GRU Iterative Updates. Fig. 4 illustrates the iterative estimation of depth maps. For better visualization, we put the reference images and the ground truth depths on the first two rows. The bottom 4 rows show the depth predictions at iteration step $t = 0, 4, 8, 12$ for each batch sample. *Itr-0* means the softargmin-start we introduced to accelerate the GRU training and convergence. We can see the depth maps are progressively improved within T iterations (here $T = 12$ in network training for the trade-off between the memory consumption and depth accuracy).

Network Inference. For inference, we set the GRU iteration number as $T = 24$ by default, and we also ablate other values of T in the main paper. The input image is in 320×256 resolution, and it is upsampled to 640×320 for ScanNet benchmark evaluation and cross-dataset generalization. The GPU memory consumption is 2088MiB from *nvidia-smi*, and runtime in inference mode is 8.6 fps when processing frames with dimension 320 ×256.

5. Additional Ablation Studies

We introduce more ablation studies to verify our design.

Frame Sampling: We compare the simple view selection strategy (i.e., sampling by every 10 frames), with the heuristics introduced in [4]. Tab. 3 shows that our methods can be further improved when the selected views have more overlapping and the baselines between them are suitable. Our(+pose,atten) even with simple strategy outperforms other variants with heuristic sampling, and so are our(+pose) vs our(base), verifying the effectiveness of each module.

Different Depth Binning. When implementing plane-sweep stereo [2, 6] to construct the cost volume, we need to sample M_0=64 plane hypotheses. In our main experiments, we use the inverse depth bins, *i.e.*, the plane hypotheses are uniformly sampled in the inverse depth space, s.t. $1/d \sim U(d_{\text{min}}, d_{\text{max}})$. Here we set d_{min}=0.25 and d_{max}=20 meters for indoor scenes (e.g., ScanNet [3]). We also test linear depth bins, i.e., $d \sim U(d_{\text{min}}, d_{\text{max}})$, and hand-crafted depth bins by calculating the depth distribution on ScanNet. But we found that inverse depth binning achieves the best results, as we reported in the main paper. We also test

[1] only the ground truth pose is used for feature warping and cost volume construction.

Method	ScanNet Test-Set							
	abs-rel (↓)	abs(↓)	sq-rel(↓)	rmse(↓)	rmse-log(↓)	$\delta < 1.25/1.25^2/1.25^3$ (↑)		
MVSNet [17]	0.1032	0.1865	0.0465	0.2743	0.1385	0.8935	0.9775	0.9942
MVSNet(+pose)	**0.0937**	**0.1714**	**0.0401**	**0.2565**	**0.1300**	**0.9072**	**0.9803**	**0.9947**
	(0.0955)	(0.1766)	(0.0431)	(0.2654)	(0.1339)	(0.9021)	(0.9785)	(0.9941)
MVSNet(+atten)	0.1018	0.1853	0.0468	0.2734	0.1377	0.8957	0.9779	0.9941
IterMVS [15]	0.0991	0.1818	0.0518	0.2733	0.1368	0.8995	0.9741	0.9915
IterMVS(+pose)	0.0958	0.1813	0.0480	0.2715	0.1343	0.9004	0.9758	0.9923
	(0.0943)	(0.1777)	(0.0472)	(0.2687)	(0.1336)	(0.9037)	(0.9764)	(0.9923)
IterMVS(+atten)	**0.0920**	**0.1741**	**0.0431**	**0.2620**	**0.1298**	**0.9066**	**0.9785**	**0.9936**

(a) Quantitative results on ScanNet Test Set [3].

Method	DTU Test-Set							
	abs-rel (↓)	abs(↓)	sq-rel(↓)	rmse(↓)	rmse-log(↓)	$\delta < 1.25/1.25^2/1.25^3$ (↑)		
MVSNet [17]	0.0143	10.7235	1.4193	25.3989	0.0356	0.9882	0.9984	1.0
MVSNet(+pose)	0.0151	11.1539	1.2867	24.3420	0.0337	0.9907	**0.9988**	1.0
	(0.0129)	(9.8094)	(1.2638)	(23.8917)	(0.0330)	(0.9905)	(0.9987)	(1.0)
MVSNet(+atten)	**0.0123**	**9.1150**	**1.1311**	**22.3525**	**0.0311**	**0.9909**	0.9986	**1.0**
IterMVS [15]	0.0146	10.6225	2.1377	28.7009	0.0404	0.9832	0.9960	0.9997
IterMVS(+pose)	**0.0129**	9.9510	**1.8261**	28.1695	0.0385	0.9831	**0.9978**	**0.9999**
	(0.0128)	(9.8926)	(1.8216)	(28.1242)	(0.0384)	(0.9832)	(0.9977)	(0.9999)
IterMVS(+atten)	0.0130	**9.4121**	1.8775	**25.6287**	**0.0357**	0.9860	0.9969	0.9993

(b) Quantitative results on DTU Test Set [8].

Table 1. Quantitative evaluation results on the test set of ScanNet [3] and DTU [8] for our modules applied to baseline MVSNet [17] and IterMVS [15]. Error metrics (lower is better) are abs-rel, abs, sq-rel, rmse, rmse-log, while accuracy (higher is better) metrics are $\delta < 1.25/1.25^2/1.25^3$. *Bold* is the best score, and *underline* indicates the second best one. The results given in parenthesis and highlighted by gray, denote that the residual pose is only used for network training, but not for inference. They are listed for reference but not for comparison with other rows.

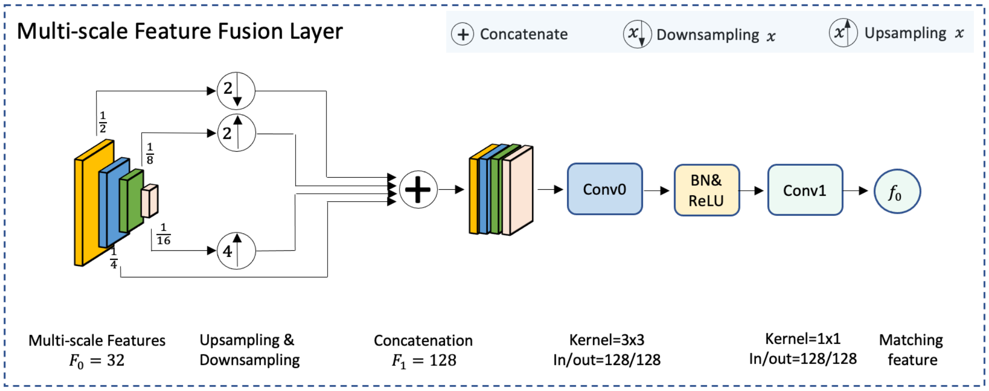
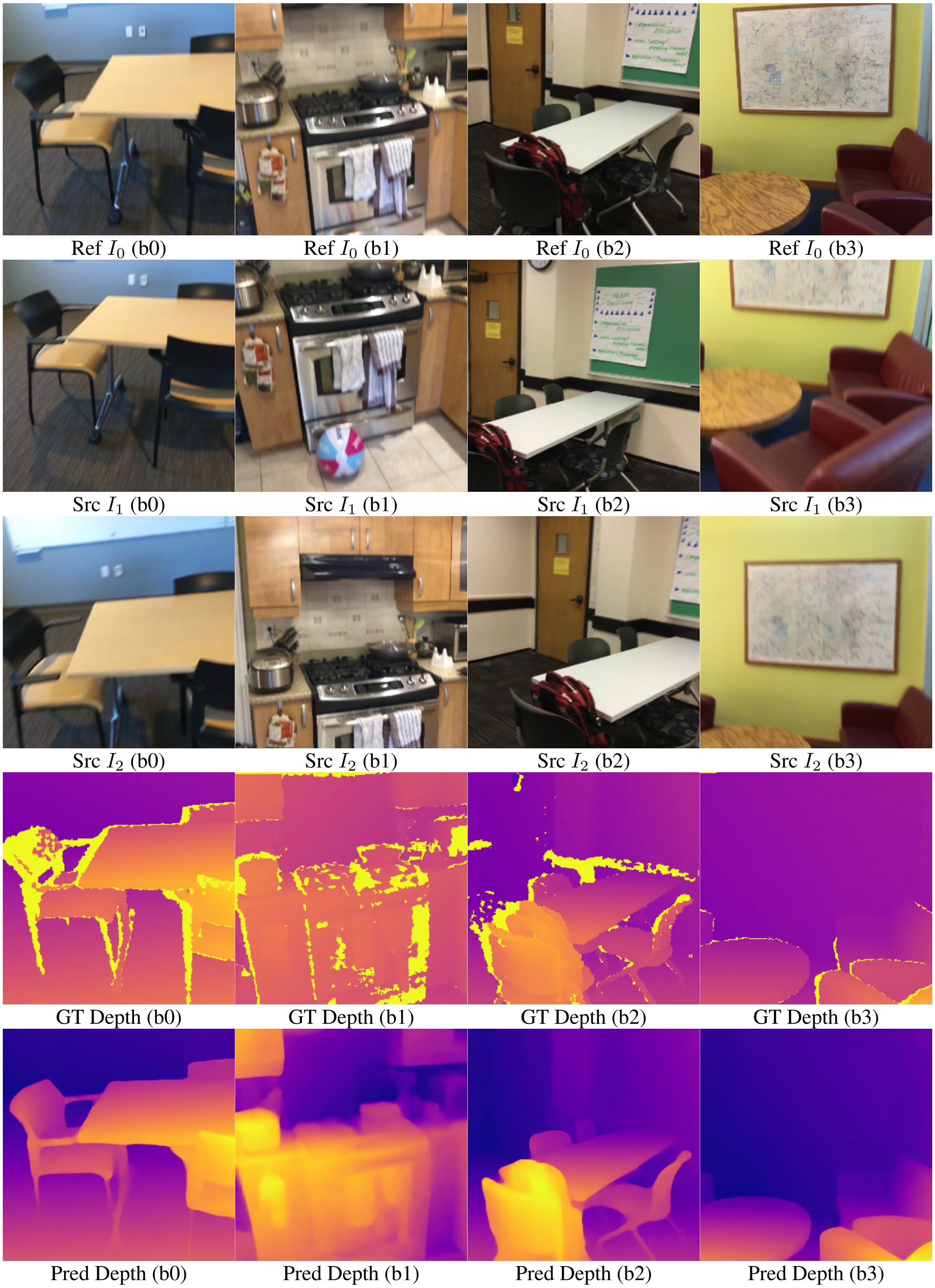


Figure 1. Multi-scale feature fusion layer.

adaptive depth bins as in [1], where the depth bins are dynamically generated upon the global feature learned by a transformer layer. For our(+pose) variant, adaptive depth bins lead to marginal improvement than the inverse depth bins. However, for our(+pose,atten) variant, adaptive depth bins hinder the depth accuracy.

6. Qualitative Results

Depth and Error Maps. More qualitative results of depth maps and error maps on the Scan-



Ref I_0 (b0) Ref I_0 (b1) Ref I_0 (b2) Ref I_0 (b3)

Src I_1 (b0) Src I_1 (b1) Src I_1 (b2) Src I_1 (b3)

Src I_2 (b0) Src I_2 (b1) Src I_2 (b2) Src I_2 (b3)

GT Depth (b0) GT Depth (b1) GT Depth (b2) GT Depth (b3)

Pred Depth (b0) Pred Depth (b1) Pred Depth (b2) Pred Depth (b3)

Figure 2. Training logs at last logging step on ScanNet [3] training set. Columns show samples and results of mini-batch ones b0, b1, b2, and b3. For the training logs, we show the color maps of the ground truth depths and predictions in the inverse space (i.e., disparity), so as to better align with the training loss calculated on the inverse depth domain.

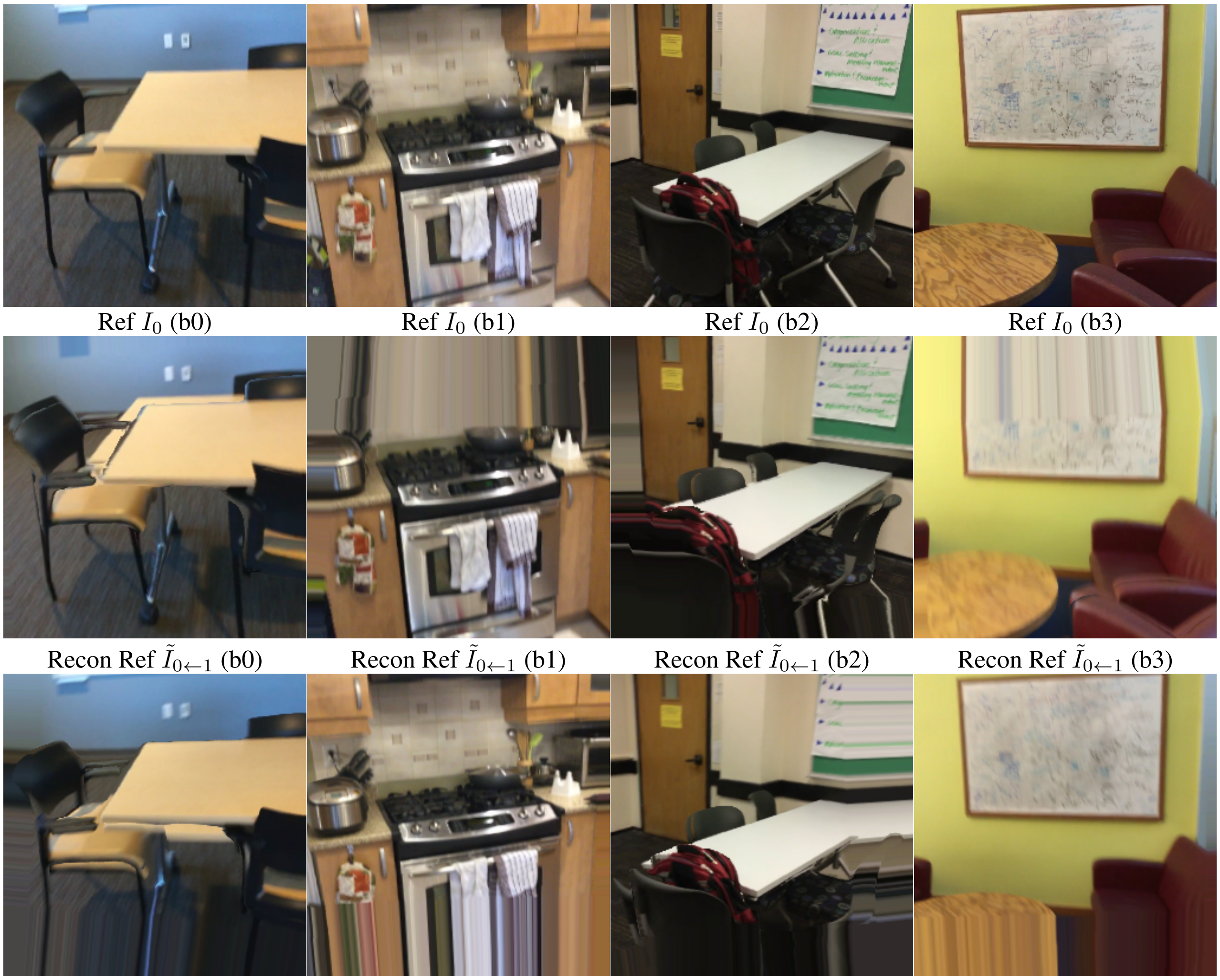
Layers	F-Net	C-Net	Transformer	Residual Pose Net	GRUs	Total
Parameter (M)	2.9545	4.3212	0.3438	13.0120	6.9501	27.5816
Percentage	10.70%	15.67%	1.25%	47.18%	25.20%	100%

(a) Our model capacity (full version).

Layers	F-Net	C-Net	Transformer	Residual Pose Net	GRUs	Total
Parameter (M)	2.9545	4.3212	0.3438	-	6.9501	14.5696
Percentage	20.28%	29.66%	2.36%	-	47.70%	100%

(b) Our model capacity, if without residual pose net.

Table 2. Our model capacity. Parameter numbers are given in million (M) and the percentage of each module is listed.



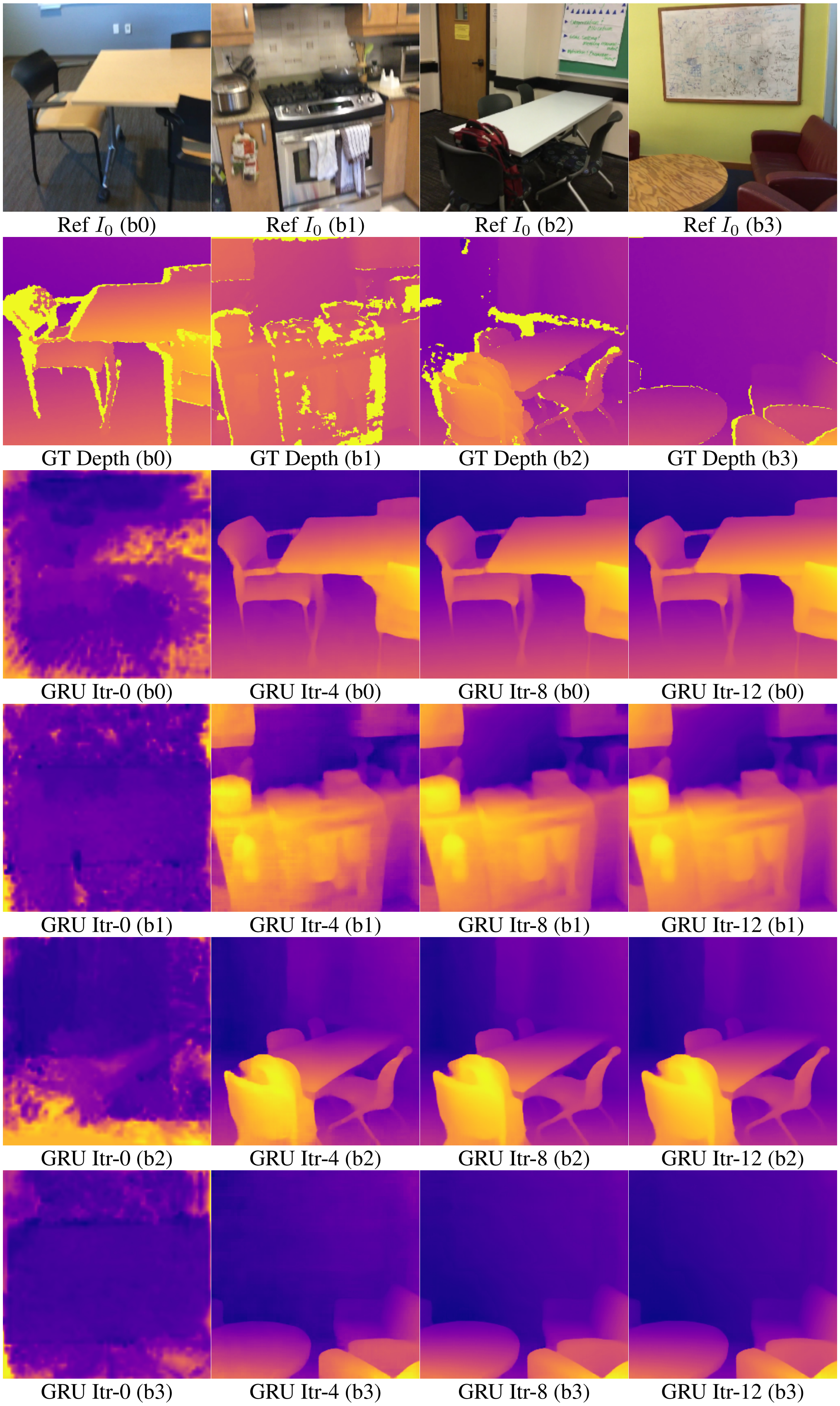
Ref I_0 (b0) Ref I_0 (b1) Ref I_0 (b2) Ref I_0 (b3)

Recon Ref $\tilde{I}_{0\leftarrow 1}$ (b0) Recon Ref $\tilde{I}_{0\leftarrow 1}$ (b1) Recon Ref $\tilde{I}_{0\leftarrow 1}$ (b2) Recon Ref $\tilde{I}_{0\leftarrow 1}$ (b3)

Recon Ref $\tilde{I}_{0\leftarrow 2}$ (b0) Recon Ref $\tilde{I}_{0\leftarrow 2}$ (b1) Recon Ref $\tilde{I}_{0\leftarrow 2}$ (b2) Recon Ref $\tilde{I}_{0\leftarrow 2}$ (b3)

Figure 3. Residual pose training. The top row shows the reference images, and the bottom two rows show the reconstructed images of the reference view by warping the source images with the updated poses and predicted depth map of the reference view.

Net test set [3] are shown in Fig. 6. The samples shown here are scene0711_00/001050.png, scene0711_00/002530.png, scene0727_00/001260.png, and scene0769_00/000720.png. The error maps contain



Ref I_0 (b0) Ref I_0 (b1) Ref I_0 (b2) Ref I_0 (b3)

GT Depth (b0) GT Depth (b1) GT Depth (b2) GT Depth (b3)

GRU Itr-0 (b0) GRU Itr-4 (b0) GRU Itr-8 (b0) GRU Itr-12 (b0)

GRU Itr-0 (b1) GRU Itr-4 (b1) GRU Itr-8 (b1) GRU Itr-12 (b1)

GRU Itr-0 (b2) GRU Itr-4 (b2) GRU Itr-8 (b2) GRU Itr-12 (b2)

GRU Itr-0 (b3) GRU Itr-4 (b3) GRU Itr-8 (b3) GRU Itr-12 (b3)

Figure 4. Iterative depth estimation from GRU layers. The bottom 4 rows show the depth predictions at iteration step $t = 0, 4, 8, 12$ for each batch sample (b1, b2, b3 and b4).

Sampling	abs-rel	abs	$\delta < 1.25$
s10 (base)	0.0885	0.1605	0.9211
key (base)	0.0838	0.1598	0.9277
s10 (+pose)	0.0827	0.1523	0.9277
key (+pose)	0.0789	0.1531	0.9339
s10 (+pose,atten)	$\underline{0.0747}$	$\underline{0.1392}$	$\underline{0.9382}$
key (+pose,atten)	**0.0697**	**0.1348**	**0.9472**

Table 3. Frame sampling comparison. The results are evaluated on the ScanNet test set [3].

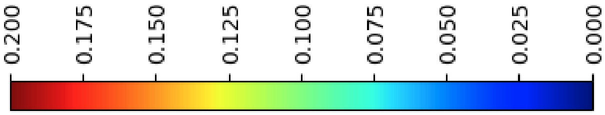


Figure 5. Color scale used for all *abs* error in depth maps in the supplementary material.

the absolute errors *abs* in depth. For the ground truth depth maps and the error maps, invalid regions (i.e., without ground truth depth annotation) are filled in black. The color maps of the ground truth depths and predictions are shown in the depth space (i.e., not in disparity space). The *abs* errors (in meters) are superimposed on the error maps for better comparison. The corresponding color bar to visualize the error maps is shown in Fig. 5. Comparing the depth predictions and the error maps for our method and the baseline IterMVS [15] and baseline PairNet [4], our method predicts more accurate estimates, especially in the challenging regions, *e.g.*, the boundary, the ground, the white wall, and the round desk.

Cross-Dataset Generalization from ScanNet to DTU Fig. 7 shows the depth maps of DTU dataset when generalized from ScanNet without fine-tuning, and our method outperforms IterMVS visibly, and on par with PairNet.

The ScanNet test set in our experiments contains 20,668 samples. As shown in Fig. 8, we report *abs* error curves (by plotting values in meters every 100 frames, out of those 20,668 samples) to reflect the distribution of the errors. We also compare the mean and standard deviation to reflect the overall performance of our method versus the baselines: mean error 0.139 (our) $<$ 0.171 (PairNet) $<$ 0.182 (IterMVS), and standard deviation 0.115 (our) $<$ 0.135 (IterMVS) $<$ 0.148 (PairNet), showing that our method consistently outperforms the baselines with smaller average and lower standard deviation.

7. Quantitative Metrics

We use the metrics defined in [5], including mean absolute error (abs), mean absolute relative error (abs-rel), squared relative error (sq-rel), RMSE in linear (rmse) and log (rmse-log) scales, and inlier ratios under thresholds of 1.25/1.25^2/1.25^3. For a predicted depth map y and ground truth y^*, each with n pixels indexed by i, those metrics are formulated as:

$$\text{abs}: \frac{1}{n}\sum_i |y_i - y_i^*|$$

$$\text{abs-rel}: \frac{1}{n}\sum_i |y_i - y_i^*|/y_i^* \quad \text{sq-rel}: \frac{1}{n}\sum_i \|y_i - y_i^*\|^2/y_i^*$$

$$\text{rmse}: \sqrt{\frac{1}{n}\sum_i \|y_i - y_i^*\|^2}$$

$$\text{rmse-log}: \sqrt{\frac{1}{n}\sum_i \|\log y_i - \log y_i^*\|^2}$$

$$\text{inlier ratio\% of } y_i \quad \text{s.t. } \max\left(\frac{y_i}{y_i^*}, \frac{y_i^*}{y_i}\right) = \delta < 1.25^i,$$

$$\text{where } i = 1, 2 \text{ and } 3.$$

References

[1] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. AdaBins: Depth estimation using adaptive bins. In *CVPR*, pages 4009–4018, 2021. 3

[2] R. T. Collins. A space-sweep approach to true multi-image matching. In *CVPR*, 1996. 2

[3] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 1, 2, 3, 4, 5, 7, 8, 10

[4] Arda Düzçeker, Silvano Galliani, Christoph Vogel, Pablo Speciale, Mihai Dusmanu, and Marc Pollefeys. DeepVideoMVS: Multi-view stereo on video with recurrent spatio-temporal fusion. In *CVPR*, 2021. 1, 2, 7, 8, 9, 10

[5] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NeurIPS*, volume 27, 2014. 1, 7

[6] David Gallup, Jan-Michael Frahm, Philippos Mordohai, Qingxiong Yang, and Marc Pollefeys. Real-time plane-sweeping stereo with multiple sweeping directions. In *CVPR*, pages 1–8, 2007. 2

[7] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-time rgb-d camera relocalization. In *International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, October 2013. 1

[8] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *CVPR*, pages 406–413, 2014. 1, 2, 3, 9

[9] Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. Learning to estimate hidden motions with global motion aggregation. In *ICCV*, pages 9772–9781, 2021. 2

[10] Kevin Lai, Liefeng Bo, and Dieter Fox. Unsupervised feature learning for 3d scene labeling. In *ICRA*, pages 3050–3057. IEEE, 2014. 1

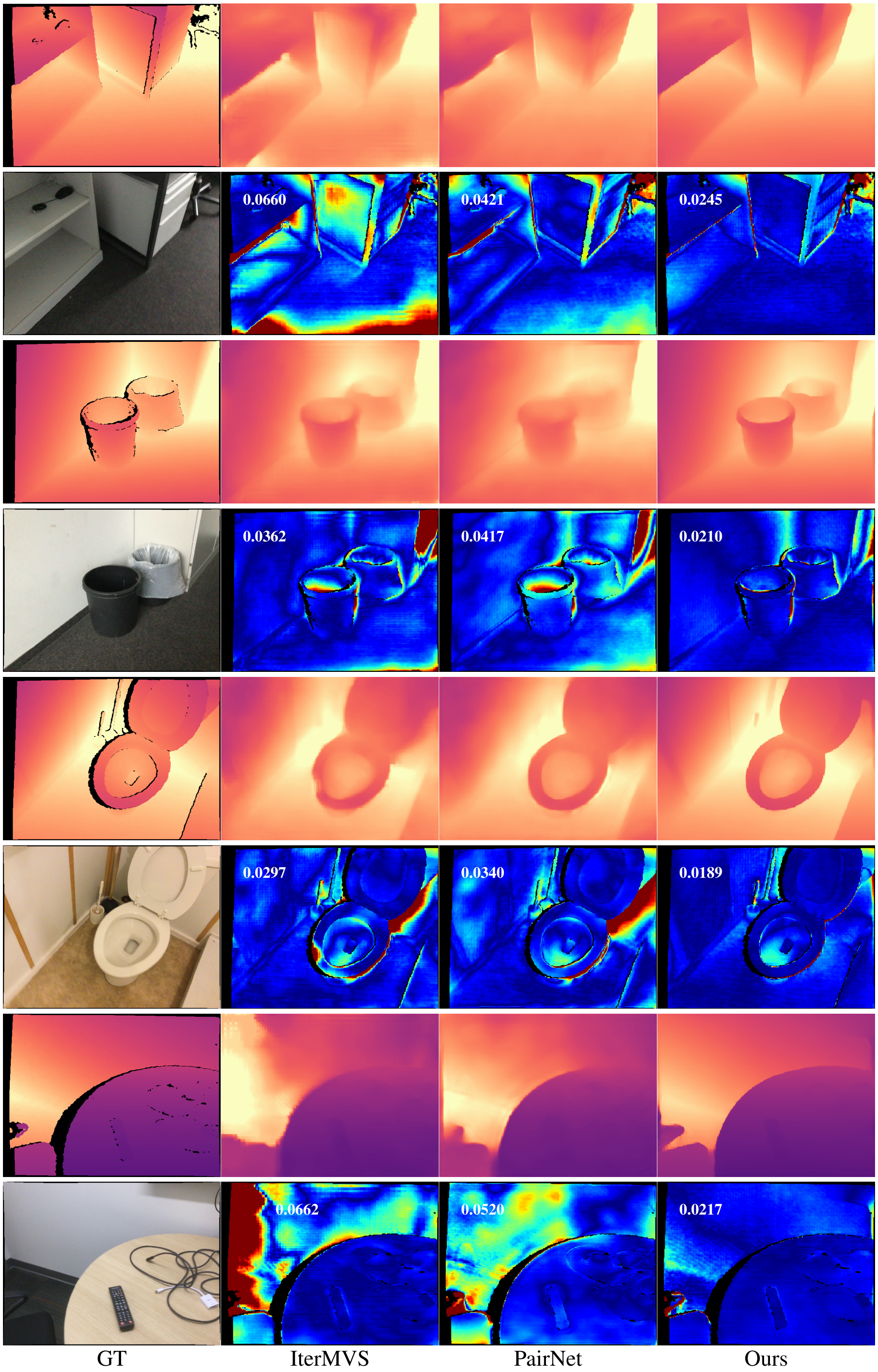


Figure 6. Qualitative results on ScanNet [3] test set. Every two rows show depth maps (top) and error maps (bottom) for a sample. The leftmost column shows ground truth depths and reference images. Others columns are the depth predictions and error maps, by IterMVS [15], PairNet [4] and ours, respectively. The abs-err errors (in meters) are imposed on the depth maps for comparison.

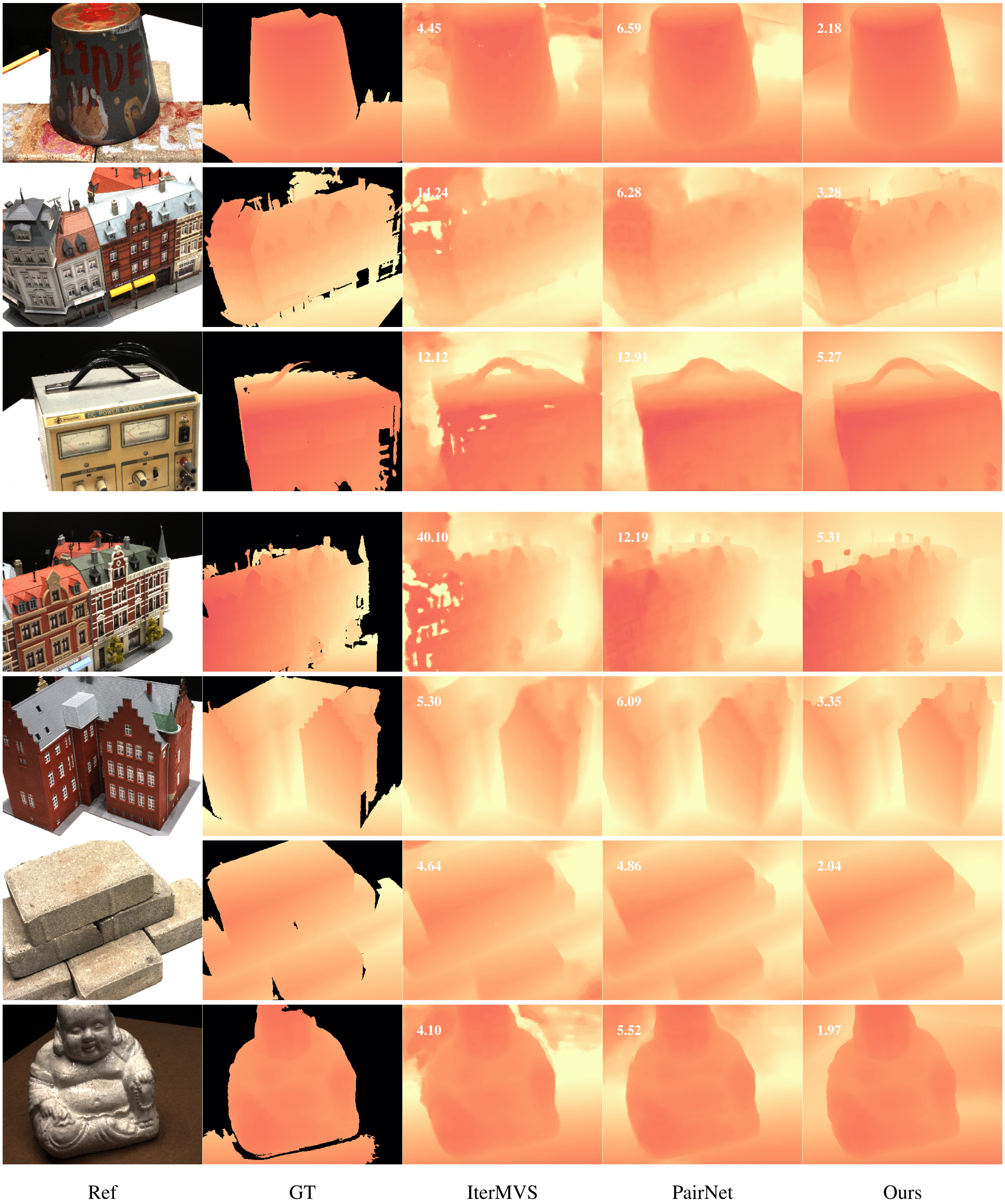


Figure 7. Cross-dataset generalization qualitative results on DTU [8] trained on ScanNet. Columns from left to right show reference image, ground truth depth, and the estimated depth for baseline IterMVS [15], PairNet [4] and our method, respectively. Our method outperforms IterMVS visibly, and on par with PairNet. The abs-err errors (in millimeters) are imposed on the depth maps for comparison.

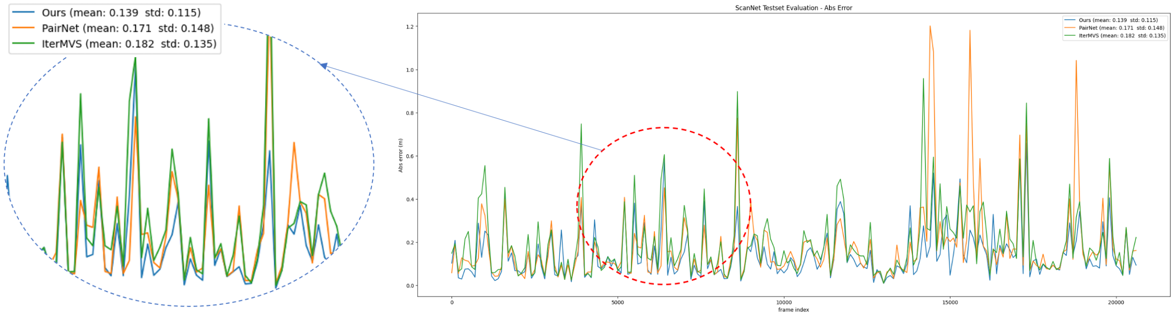


Figure 8. Absolute error metric curves evaluated on all the frames of ScanNet [3] test set, for our method and baselines IterMVS [15] and PairNet [4]. Please enlarge the figures to better view the metrics and legends displayed in the top-left corner.

[11] Lahav Lipson, Zachary Teed, and Jia Deng. Raft-Stereo: Multilevel recurrent field transforms for stereo matching. In *3DV*, pages 218–227. IEEE, 2021. 1, 2

[12] Xiaoxiao Long, Lingjie Liu, Wei Li, Christian Theobalt, and Wenping Wang. Multi-view depth estimation using epipolar spatio-temporal networks. In *CVPR*, pages 8258–8267, June 2021. 1

[13] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8024–8035, 2019. 1

[14] Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *ECCV*, pages 402–419. Springer, 2020. 2

[15] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, and Marc Pollefeys. IterMVS: Iterative probability estimation for efficient multi-view stereo. In *CVPR*, 2022. 1, 2, 3, 7, 8, 9, 10

[16] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. PatchmatchNet: Learned multi-view patchmatch stereo. In *CVPR*, 2021. 1

[17] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. MVSNet: Depth inference for unstructured multi-view stereo. *ECCV*, 2018. 1, 2, 3

[18] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent MVSNet for high-resolution multi-view stereo depth inference. *CVPR*, 2019. 1