Supplementary Material of When Epipolar Constraint Meets Non-local Operators in Multi-View Stereo

Tianqi Liu Xinyi Ye Weiyue Zhao Zhiyu Pan Min Shi* Zhiguo Cao
Key Laboratory of Image Processing and Intelligent Control, Ministry of Education; School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China
{tianqiliu,xinyiye,zhaoweiyue,zhiyupan,min_shi,zgcao}@hust.edu.cn

A. More Details of Epipolar Pair Search

A.1. Mathematical Derivation

In Sec. 4.1, we obtain the expression of the corresponding epipolar lines (Eq. 3) from point homography transformation (Eq. 1). Here, we will provide a detailed derivation of the process. To keep consistent, we illustrate the mathematical derivation with one source view.

Given a pixel $\mathbf{p}_r = (x_r, y_r, 1)^T$ in the reference view, the corresponding pixel $\mathbf{p}_s$ in the source view is

$$\mathbf{p}_s(d) = \mathbf{K}_s[\mathbf{R}(\mathbf{K}_r^{-1}\mathbf{p}_r d) + \mathbf{t}]\,, \qquad \text{(a1)}$$

where d denotes the depth of the reference pixel; $\mathbf{R}$ and $\mathbf{t}$ indicate the rotation and translation matrices between the reference and the source view; $\mathbf{K}_r$ and $\mathbf{K}_s$ are the intrinsic matrices of the reference view and source view, respectively. For easy understand, Eq. (a1) can be expressed as

$$\mathbf{p}_s(d) = d_s * (x_s(d), y_s(d), 1)^T = \mathbf{W}\mathbf{p}_r d + \mathbf{b}\,, \qquad \text{(a2)}$$

where $\mathbf{W} = \mathbf{K}_s\mathbf{R}\mathbf{K}_r^{-1}$, $\mathbf{b} = \mathbf{K}_s\mathbf{t}$. We can further transform Eq. (a2) into a coordinate form:

$$x_s(d) = \frac{a_1 d + b_1}{a_3 d + b_3}\,, y_s(d) = \frac{a_2 d + b_2}{a_3 d + b_3}\,, \qquad \text{(a3)}$$

where $a_1 = w_{11}x_r + w_{12}y_r + w_{13}$, $a_2 = w_{21}x_r + w_{22}y_r + w_{23}$, $a_3 = w_{31}x_r + w_{32}y_r + w_{33}$; w_{ij} is an element of matrix $\mathbf{W}$, and b_i is an element of vector $\mathbf{b}$.

Since $\{a_i\}_{i=1}^3$ and $\{b_i\}_{i=1}^3$ are constants associated with the camera parameters and the coordinate of $\mathbf{p}_r$, the standard equation for the epipolar line $y_s(d) = kx_s(d) + b$ can be formulated as

$$\begin{cases} k = \dfrac{\Delta y_s(d)}{\Delta x_s(d)} = \dfrac{a_2 b_3 - a_3 b_2}{a_1 b_3 - a_3 b_1} \\ b = y_s(0) - kx_s(0) = \dfrac{b_2}{b_3} - k\dfrac{b_1}{b_3} \end{cases}. \qquad \text{(a4)}$$

Specifically, when $\Delta x_s(d) \to 0$, the stand equation for the epipolar line is $x_d(s) = k^{'} y_s(d) + b^{'}$, which can be formulated as

$$\begin{cases} k^{'} = \dfrac{\Delta x_s(d)}{\Delta y_s(d)} = \dfrac{a_1 b_3 - a_3 b_1}{a_2 b_3 - a_3 b_2} \\ b^{'} = x_s(0) - k^{'} y_s(0) = \dfrac{b_1}{b_3} - k^{'}\dfrac{b_2}{b_3} \end{cases}. \qquad \text{(a5)}$$

A.2. Discussion about Quantification

we quantify the pre-calculated k and b by rounding as

$$\begin{cases} k = s_k * \text{round}(\dfrac{k}{s_k}) \\ b = s_b * \text{round}(\dfrac{b}{s_b}) \end{cases}, \qquad \text{(a6)}$$

where s_k and s_b are the hyperparameters for rounding, and quantization precision (k, b) depends on the precision of them. To explore the effect of quantization precision for epipolar pair search, we list common precision combinations of s_k and s_b in Table A1.

The quantization precision of both k and b will affect the effect of epipolar pair search, and thus affect the performance. Besides, the quantization precision also influences the efficiency of epipolar pair search, as finer quantization precision leads to more clusters and vice versa. Considering the effectiveness and efficiency, we choose the precision combination of $s_k = 0.1$ and $s_b = 10$ in our implementation.

B. Efficiency Comparison

We empirically study the efficiency of the point-to-line and the line-to-line implementations in our ablation studies (Sec. 5.3). Here, we analyze their complexity theoretically. In addition, we compare the global aggregation strategy: plane-to-plane (Linear), which is applied in TransMVSNet [2].

*Corresponding author

s_k	s_b	ACC.(mm) ↓	Comp.(mm) ↓	Overall(mm) ↓
1	0.1	0.327	0.267	0.297
1	1	0.328	0.260	0.294
1	10	0.325	0.263	0.294
0.1	0.1	**0.324**	0.263	0.294
0.1	1	0.325	0.257	**0.291**
0.1	10	0.329	**0.253**	**0.291**
0.01	0.1	0.325	0.271	0.298
0.01	1	0.332	0.260	0.296
0.01	10	0.327	0.265	0.296

Table A1. **Comparison of different quantification precision.**

B.1. Theoretical Efficiency Comparison

Given $Q \in R^{B\times N_1\times C}$, $K \in R^{B\times N_2\times C}$, and $V \in R^{B\times N_2\times C}$, the computational complexity of the vanilla Transformer [7] is $B(9N_1C^2 + 2N_2C^2 + 2N_1N_2C)$. The computational complexity of the linear Transformer [3] is $B(10N_1C^2 + 3N_2C^2)$.

For line-to-line and point-to-line, the computational complexity depends on the number of epipolar lines as well. Specifically, suppose there are M corresponding epipolar lines and the average number of pixels on an epipolar line is S. For point-to-line, $B = HW$, $N_1 = 1$, $N_2 = S$, its computational complexity is $HW(9C^2 + 2SC^2 + 2SC)$. For line-to-line, $B = M$, $N_1 = S$, $N_2 = S$, its computational complexity is $M(11SC^2 + 2S^2C)$. It is worth noting that M and S are of the same order of magnitude as H and W. They are usually smaller because the epipolar lines only exist in the common view of the two images. For the plane-to-plane in the form of linear Transformer implementation, $B = 1$, $N_1 = N_2 = HW$, its computational complexity is $13HWC^2$.

For a more intuitive comparison, we set $H = 80$, $W = 64$, $C = 64$, $S = 30$, $M = 30$: the computational complexity of point-to-line is 1.5G; the computational complexity of line-to-line is 0.04G; and the computational complexity of plane-to-plane (linear) is 0.27G.

B.2. Empirical Efficiency Comparison

We report the inference time to compare different aggregate ways in practice. As shown in Table A2, “line-to-line” is a more efficient and effective way to aggregate information, which maintains the highest performance while being the lowest in terms of time and memory consumption.

C. Additional Ablation Studies

C.1. Number of Blocks

Table A3 shows the impact of different block numbers of the Intra-Epipolar Augmentation (IEA) and Cross-Epipolar Augmentation (CEA). As the number increases, no performance gain is obtained, which suggests that one block is sufficient for non-local feature aggregation. Since more blocks lead to larger computation overhead, N_a is set to 1 in our implementation.

mothod	Overall(mm) ↓	Time(ms) ↓	Memory(MB) ↓
Line-to-line	0.291	**2.0**	**2769**
Point-to-line	**0.290**	3.5	4207
Plane-to-plane (Linear)	0.303	3.9	2997

Table A2. **Comparison of different ways of information aggregation.** “Line-to-line” refers to the information aggregation between epipolar pairs. “Point-to-line” refers to a point interacting with its corresponding epipolar line. “Plane-to-plane (Linear) refers to the information aggregation between two whole images in the form of linear Transformer [3] implementation. “Time” refers to inference time through one Transformer (for 864 × 1152 images).

N_a	ACC.(mm) ↓	Comp.(mm) ↓	Overall(mm) ↓	Time(s) ↓	Param(M) ↓
1	0.329	**0.253**	0.2910	**0.46**	**1.09**
2	**0.327**	0.254	**0.2905**	0.47	1.16
3	**0.327**	0.254	**0.2905**	0.47	1.23

Table A3. **Ablation study on the number of IEA and CEA blocks.**

PE	ACC.(mm) ↓	Comp.(mm) ↓	Overall(mm) ↓
w/o	**0.329**	0.260	0.295
learnable	0.330	0.254	0.292
sine	**0.329**	**0.253**	**0.291**

Table A4. **Ablation study on spatial positional encoding.**

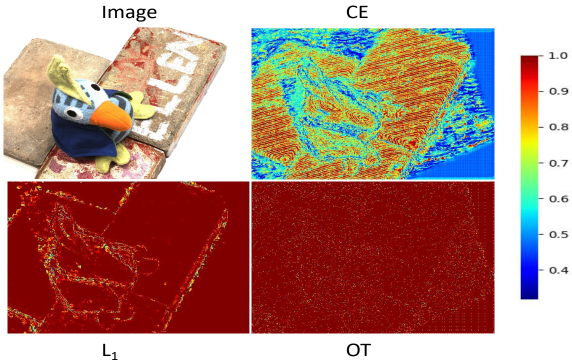


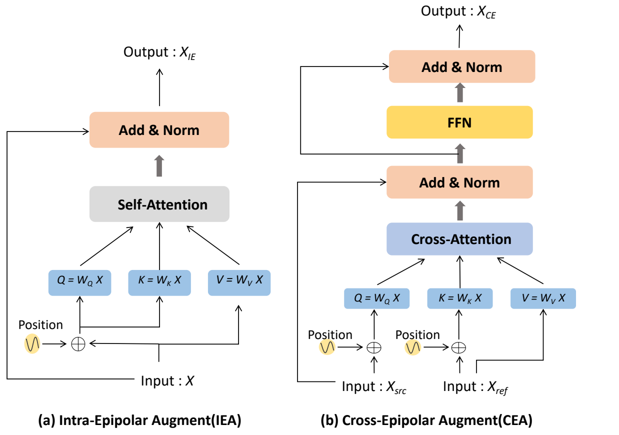
Figure A1. **Visualization of confidence maps generated through different loss functions.**

C.2. Spatial Positional Encoding

In Epipolar Transformer (ET), we apply Positional Encoding (PE) to add spatial positional information to feature sequences. We compare different positional encoding implementations in Table A4. Positional encoding is necessary and the performance of “learnable” and “sine” positional encoding are similar. Since “sine” positional encoding is parameter-free, we used “sine” positional encoding in our implementation.

Loss	ACC.(mm) ↓	Comp.(mm) ↓	Overall(mm) ↓
CE	**0.329**	0.253	**0.291**
OT	0.383	**0.223**	0.303
L_1	0.367	0.245	0.306

Table A5. **Ablation study on different loss functions.**“CE” refers to the commonly used cross-entropy loss and “OT” refers to the Wasserstein loss computed by optimal transport, where the depth estimation is regarded as a classification problem. “L_1” refers to the average absolute value error where the depth estimation is regarded as a regression problem.



(a) Intra-Epipolar Augment(IEA)

(b) Cross-Epipolar Augment(CEA)

Figure A2. **The structure of IEA and CEA.** IEA is based on the self-attention mechanism while CEA is based on the cross-attention mechanism.

Order	ACC.(mm) ↓	Comp.(mm) ↓	Overall(mm) ↓
IEA+CEA	0.329	**0.253**	**0.291**
CEA+IEA	**0.327**	0.257	0.292

Table A6. **Ablation study on different orders of IEA and CEA.**

C.3. Loss Function

In Sec. 4.3, we formulate depth estimation as a classification problem and apply cross-entropy loss as the loss function of ET-MVSNet. As shown in Table A5, we compare different loss functions. The experimental results indicate that the cross-entropy loss achieves the highest overall metric, which better balances accuracy and completeness. Besides, as shown in Fig. A1, the confidence map generated through “CE” is more advantageous for filtering outliers and obtaining more accurate point clouds.

C.4. Order of IEA and CEA

As shown in Fig. A2, the Intra-Epipolar Augmentation (IEA) and Cross-Epipolar Augmentation (CEA) modules perform information aggregation within and across epipolar lines, respectively. We explore the order of IEA and CEA in Table A6. The order of IEA and CEA has little impact on the final performance. In our implementation, IEA is executed first followed by CEA.

D. Depth Map Fusion

As described in the main text, the predicted depth maps of multiple views are filtered and fused into a point cloud. Previous MVS methods always choose the suitable fusion method. In the paper, we follow the commonly used dynamic checking strategy [9] for depth filtering and fusion on both DTU dataset [1] and Tanks and Temples benchmark [4].

On the DTU dataset, we filter the confidence map of the last stage with a confidence threshold(0.55) to measure photometric consistency. For geometry consistency, we use a strict standard, as shown below.

$$err_c < thresh_c \, , err_d < log(thresh_d) \, , \tag{a7}$$

where err_c and err_d denote the reprojection coordinate error and relative error of reprojection depth, respectively. $thresh_c$ and $thresh_d$ denote thresholds for err_c and err_d, respectively. In addition, we adopt the normal(pcd) fusion method [6], and our method can achieve 0.298 on the “overall” metric.

On the Tanks and Temples benchmark, We follow [5] to adjust hyperparameters for each scene including confidence thresholds, geometric thresholds, etc. For benchmarking on the advanced set of Tanks and Temples , the number of depth hypotheses in the coarsest stage is changed from 8 to 16. And we use the model trained on the DTU dataset to reconstruct the “Horse” scene, and then use the fine-tuned model on BlendedMVS dataset [10] to reconstruct other scenes.

E. More Visualization Results

Qualitative Analysis. The Qualitative results of Tanks and Temples benchmark [4] are shown in Fig. A3. Compared with other state-of-the-art methods [8, 2, 5], our method can reconstruct more details in some challenging areas, such as surfaces with weak and repetitive textures.

Epipolar Pairs. We visualize some epipolar line pairs searched by our algorithm in Fig. A4. In the case of large differences, pixels with the same semantic information are still on the same epipolar line pair, indicating the effectiveness of our algorithm.

More Point Cloud Results. More visualization results of our model are shown in Fig. A5, which contains point clouds of Tanks and Temples benchmark [4].

References

[1] Henrik Aanaes, Rasmus Ramsbol Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for

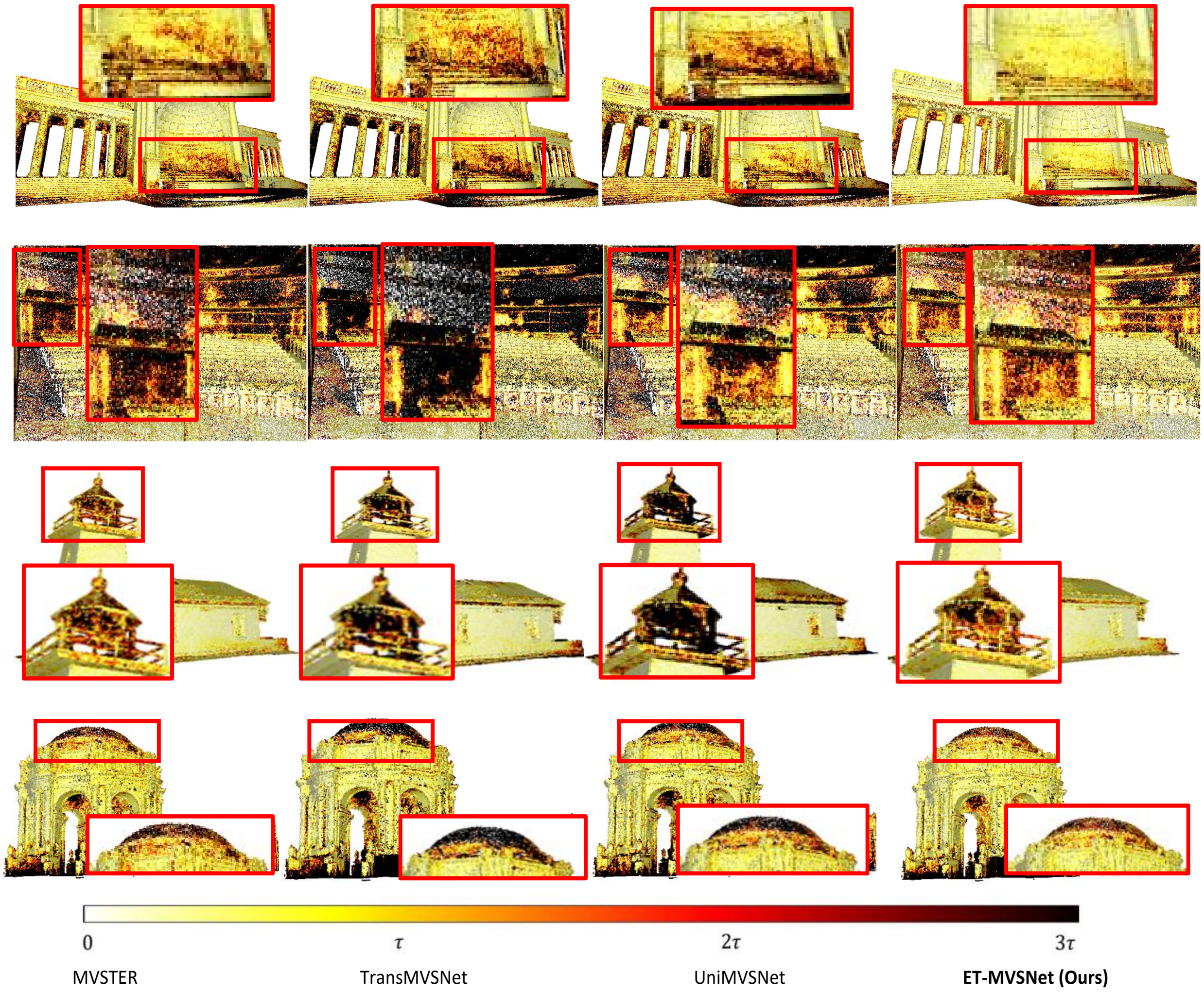


Figure A3. **Visualization comparison with state-of-the-art methods [8, 2, 5] on Tanks and Temples benchmark.** From top to bottom is the Recall of the scene of the Temple($\tau = 15mm$), Auditorium ($\tau = 10mm$), Lighthouse ($\tau = 5mm$), and Palace ($\tau = 30mm$), respectively.

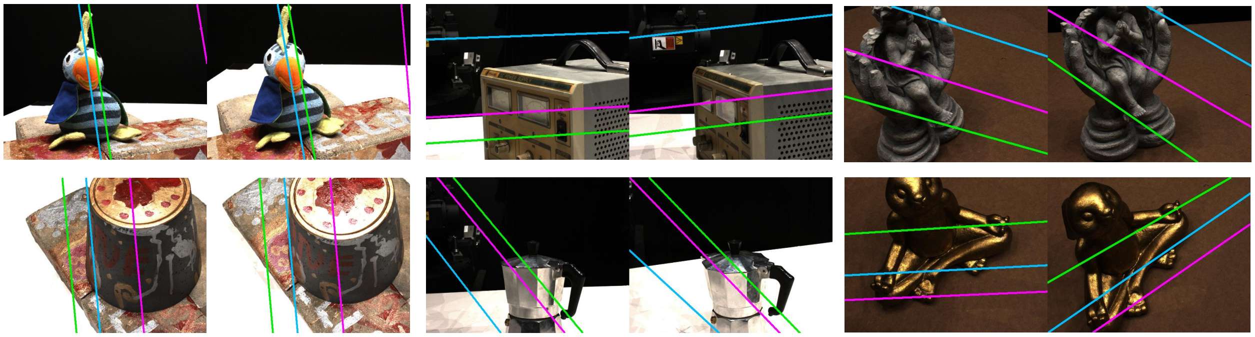


Figure A4. **Visualization of epipolar line pairs.** The same color indicates the corresponding epipolar line pair.

multiple-view stereopsis. *Int. J. Comput. Vis.*, 120:153–168, 2016.

[2] Yikang Ding, Wentao Yuan, Qingtian Zhu, Haotian Zhang, Xiangyue Liu, Yuanjiang Wang, and Xiao Liu. Transmvsnet global context-aware multi-view stereo network with transformers. In *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*,

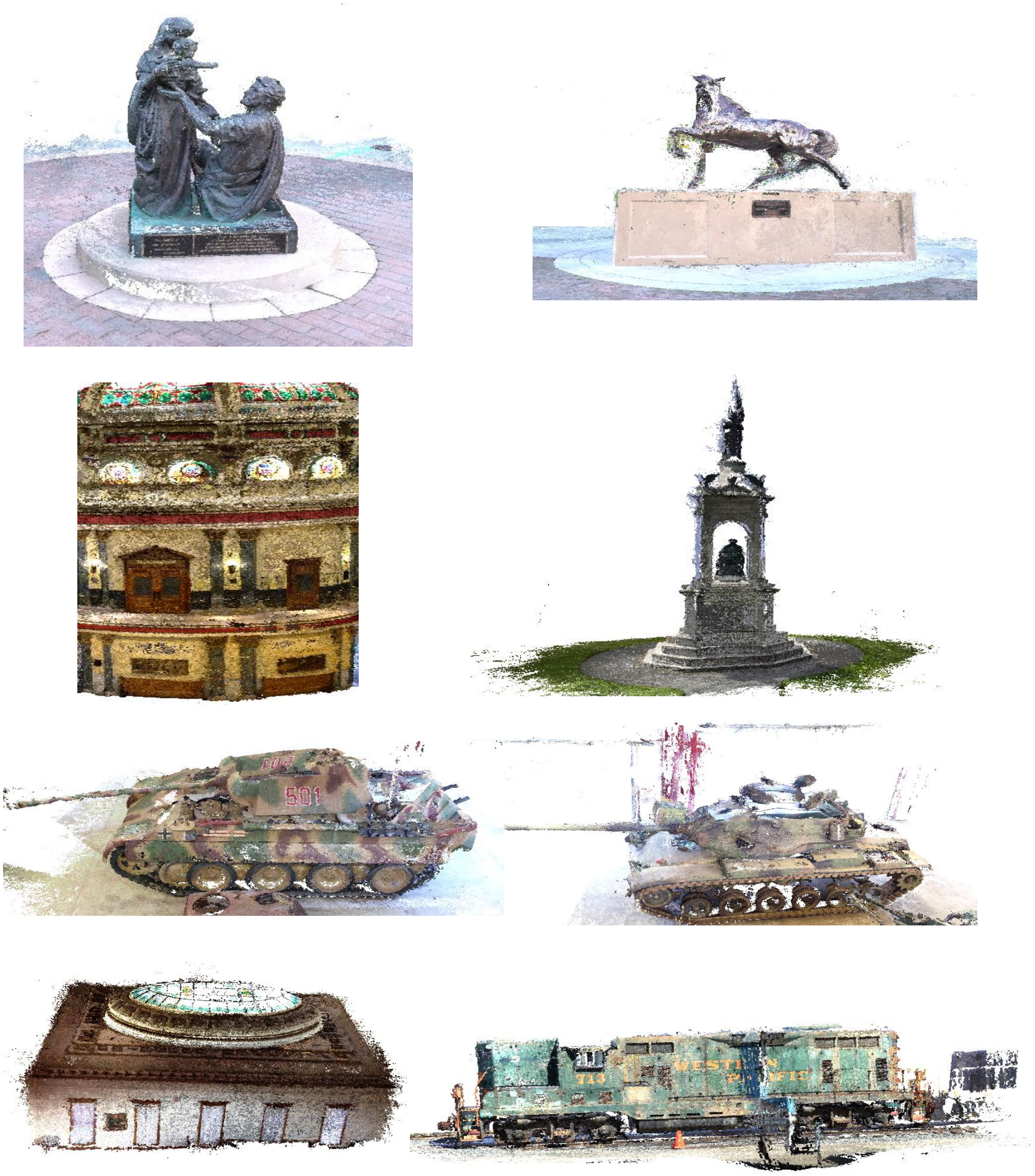


Figure A5. **Point clouds reconstructed by ET-MVSNet on Tanks and Temples benchmark [4].**

pages 8585–8594, 2022.

[3] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proc. Int. Conf. Mach. Learn.*, pages 5156–5165. PMLR, 2020.

[4] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples benchmarking large-scale scene reconstruction. *ACM Trans. Graph.*, 36(4):1–13, 2017.

[5] Rui Peng, Rongjie Wang, Zhenyu Wang, Yawen Lai, and Ronggang Wang. Rethinking depth estimation for multi-view stereo a unified representation. In *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 8645–8654, 2022.

[6] Johannes L Schonberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Proc. Eur. Conf. Comput. Vis.*, pages 501–518. Springer, 2016.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Proc. Adv. Neural Inf. Process. Syst.*, 30, 2017.

[8] Xiaofeng Wang, Zheng Zhu, Guan Huang, Fangbo Qin, Yun Ye, Yijia He, Xu Chi, and Xingang Wang. Mvster epipolar transformer for efficient multi-view stereo. In *Proc. Eur. Conf. Comput. Vis.*, pages 573–591. Springer, 2022.

[9] Jianfeng Yan, Zizhuang Wei, Hongwei Yi, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Dense hybrid recurrent multi-view stereo net with dynamic consistency checking. In *Proc. Eur. Conf. Comput. Vis.*, pages 674–689. Springer, 2020.

[10] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs a large-scale dataset for generalized multi-view stereo networks. In *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 1790–1799, 2020.