SUPPLEMENTARY MATERIAL

Anonymous ICME submission

1. IMPLEMENTATION DETAILS

The backbone of ParaSurRe is adopted from VolSDF[1], which use two networks f_ϕ and f_θ to represent radiance and Signed Distance Function (SDF) respectively (Fig. 1). We follow InstantNGP[2] and parameterize both f_θ and f_ϕ by multi-resolution feature grids and a 3-layer MLP. For each input coordinates, we interpolate multi-level features from multi-resolution feature grids, and then concatenate them and send into a shallow MLP to get SDF or radiance, as shown in Fig. 2. For SDF hash table, we set max grid level L to 8, the numbers of features per level l to 4, the coarsest resolution N_c to 16, and the finest resolution N_f to 2048. For radiance hash table, we set max grid level L to 16, the numbers of features per level l to 2, the coarsest resolution N_c to 16, and the finest resolution N_f to 2048. Density $\sigma(x)$ is modeled by a transformation of output SDF $d(x)$:

$$\sigma(x) = \alpha\Phi_\beta(-d(x)) \tag{1}$$

where Φ_β is the Cumulative Distribution Function(CDF) of the Laplace distribution with β scale and zero mean, and α, β are learnable parameters. In practice, $\alpha = \frac{1}{\beta}$. Since ParaSurRe trains in parallel, each cluster has its own learned β. To convert fused SDF to density, we compute the average of all β in different clusters and use the same transformation in Equation 1 to get density prediction. Our networks are implemented in PyTorch with Adam optimizer. In practice, to parameterize rotation R, we use the axis-angle representation $\phi = \alpha\omega$, where α denotes rotation angle and ω denotes rotation axis.

2. EVALUATION OF POSES

As mentioned in the main paper, the estimated camera trajectory are up to a 3D similarity transformation $P \in Sim(3)$.

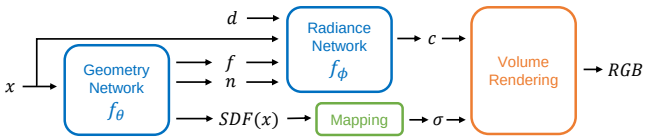

Fig. 1. Overview of the network architecture of the backbone of ParaSurRe. Our backbone is adopted from VolSDF[1].

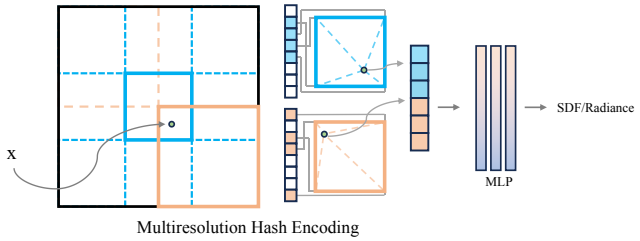

Fig. 2. Network architecture of multi-resolution feature grid.

T comprises three parameters: (s, R, T), where $R \in SO(3)$ represents rotation, $P \in R^3$ is for translation, and $s \in R$ denotes scale. We use Procrustes analysis to compute P and align estimated camera coordinates systems to ground truth camera coordinates systems. After alignment, we measure rotation distance with mean rotation error in Equation 2:

$$\theta_{error} = \frac{1}{N}\sum_{i=1}^{N}\arccos(trace(R_i^{-1}R_i^{gt})) \tag{2}$$

where N is the number of cameras, and R_i, R_i^{gt} are the estimated rotation matrix and ground truth rotation matrix respectively. In terms of translation, we use the ATE RMSE in Equation 3 to measure translation error:

$$t_{error} = \frac{1}{N}\sum_{i=1}^{N}||T_i - T_i^{gt}|| \tag{3}$$

where T_i, T_i^{gt} are the aligned translation and ground truth translation respectively.

In terms of final pose estimation, the feature matching algorithm has an impact on the final result. To make a fair comparison with Colmap[3], we use the default image matching algorithm in Colmap[3] to build scene graph. Meanwhile, note that registration order in incremental SFM also influences the final pose accuracy, we simply use the sequence order provided by Colmap[3] for fair comparison, same as default setting of Level-S^{2}fM[4].

3. ADDITIONAL RESULTS

We report more pose estimation results of Colmap, Level-S^{2}fM[4] and ParaSurRe on DTU[5] and BlendedMVS[6], as

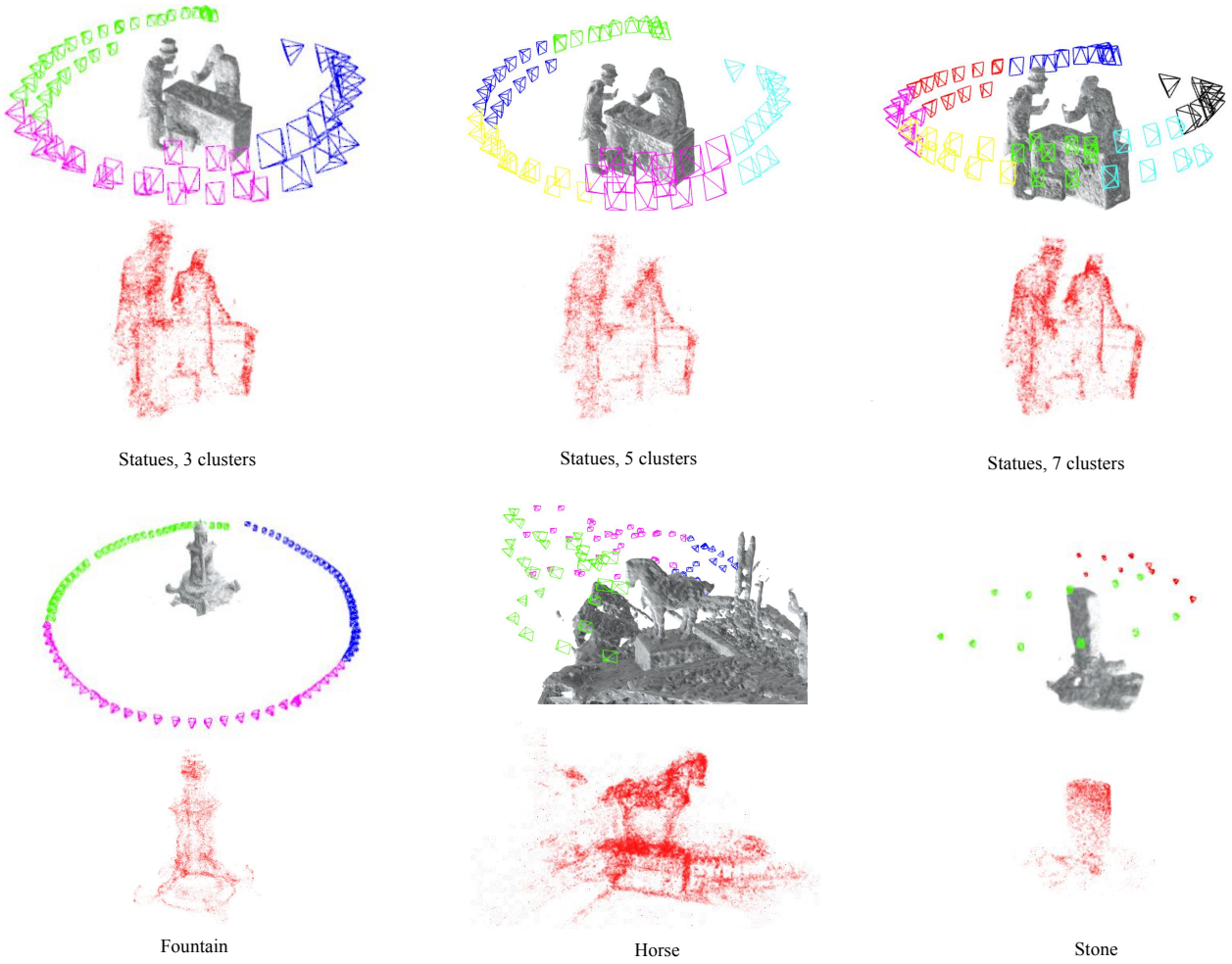

Fig. 3. Qualitative results for surface reconstruction and pose estimation of ParaSurRe. For each scene, we show the corresponding fused surface with cameras, and point clouds.

shown in Table 1 and Table 2. We also report more qualitative results for ParaSurRe in Fig. 3, where point clouds, cameras, and meshes are visualized. Different-colored cameras are located in separate clusters.

Scan	Rotation/Translation Error		
	Colmap	LevelS2FM	ParaSurRe
65	0.684/0.410	0.741/2.063	0.666/1.412
106	0.618/0.332	0.386/0.898	0.199/0.706
114	0.419/0.655	0.163/0.605	0.434/1.693
122	0.636/0.327	0.247/0.501	0.207/0.638
Mean	0.589/0.431	0.384/1.017	0.376/1.112

Table 1. Pose estimation results on DTU dataset. The best and second-best results are highlighted in purple and cyan respectively.

Scenes	Rotation/Translation Error		
	Colmap	LevelS2FM	ParaSurRe
Fountain	3.555/0.031	1.500/0.022	2.857/0.089
Stone	0.661/0.034	0.806/0.106	0.705/0.015
Statues	1.198/0.004	0.278/0.005	0.909/0.036
Horse	0.326/0.007	0.576/0.042	1.120/0.037
Mean	1.435/0.019	0.790/0.044	1.398/0.043

Table 2. Pose estimation results on BlendedMVS dataset. The best and second-best results are highlighted in purple and cyan respectively.

4. REFERENCES

[1] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman, "Volume rendering of neural implicit surfaces," *Advances in Neural Information Processing Systems*, vol. 34, pp. 4805–4815, 2021.

[2] Thomas Müller, Alex Evans, Christoph Schied, and

Alexander Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *TOG*, vol. 41, no. 4, pp. 1–15, 2022.

[3] Johannes L Schonberger and Jan-Michael Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.

[4] Yuxi Xiao, Nan Xue, Tianfu Wu, and Gui-Song Xia, "Level-s^{2}fm: Structure from motion on neural level set of implicit surfaces," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17205–17214.

[5] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs, "Large scale multi-view stereopsis evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 406–413.

[6] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1790–1799.